UNIVERSIDAD NACIONAL DE COLOMBIA

A kernel framework to support video data analysis

Hernán David Insuasti Ceballos

Universidad Nacional de Colombia
Faculty of Engineering and Architecture
Department of Electric, Electronic and Computing Engineering
Manizales, Colombia
2016

A kernel framework to support video data analysis

Hernán David Insuasti Ceballos

Thesis submitted as a partial requirement to receive the grade of:
Master in Engineering - Industrial Automation

Advisor:
Ph.D. Germán Castellanos Domínguez

Co-advisor:
Ph.D. Andrés Marino Álvarez Meza

Academic Research Group:
Signal Processing and Recognition Group - SPRG

Universidad Nacional de Colombia
Department of Electric, Electronic and Computing Engineering
Manizales, Colombia
2016

Un esquema kernel para apoyar el análisis de secuencias de video

Hernán David Insuasti Ceballos

Tesis presentada como requisito parcial para optar al título de:
Magister en Ingeniería - Automatización Industrial

Director:
Ph.D. Germán Castellanos Domínguez

Co-directorr:
Ph.D. Andrés Marino Álvarez Meza

Grupo de trabajo académico:
Grupo de Procesamiento y Reconocimiento de Señales - GPRS

Universidad Nacional de Colombia
Departamento de Ingeniería Eléctrica, Electrónica y Computación
Manizales, Colombia
2016

I would like to dedicate this work to my family; they supported me through this way, and they always have been the engine to achieve all my goals.

Acknowledgements

I would like to acknowledge the help of all involved in the collaboration process of this thesis, without whose support this work could not have been satisfactory completed. Special thanks to Professor German Castellanos, my advisor, for his orientation in this work. Also, I would like to express my gratitude to Andres Álvarez, who guided my steps in this research and his patience. A further special note of thanks goes also to all the staff at SPRG, to Santiago Molina, who was my research partner in the video applications field in the SPRG group. Finally, thanks to my family.

This work would not have been possible without the financial assistance provided by the Plataforma tecnológica para los servicios de teleasistencia, emergencias médicas, seguimiento y monitoreo permanente de pacientes y apoyo a programas de prevención project supported by Alianza Regional en TIC aplicadas - ARTICA and to the project 16882 funded by Universidad Nacional de Colombia sede Manizales and Universidad de Caldas.

Abstract

The aim of this work is to propose a kernel-based framework to support the analysis of video data. The goal is to reveal the most salient information from high-dimensional and correlated data for enhancing feature representations and easing the classification stage. In human activity recognition, the relevant patterns have nonlinear dynamics that are hidden in the data. To expose these dynamics, including prior knowledge about ideal relationships can help to find nonlinear dependence on data. In the present work, three approaches to improve the feature representation and alleviate the classification stage. Firstly, a kernel-based representation is introduced to reveal the most important relations into a codebook in a BoF approach by including of an ideal kernel of similarities between the codewords. Secondly, a methodology for feature relevance analysis is proposed. The method finds the most discriminative set of spatiotemporal features to enhance the class inter-separability by using a center kernel alignment criteria. Finally, a new method for tuning a box constraint parameter C in an SVM based on a distance between two kernels of similarities is proposed. The proposed methods are compared with the baseline techniques of the state-of-the-art showing excellent results for improving data representation and high classification rates.
Keywords: Bag of Features(BoF) , CKA, Kernel representation, SVMs, Spatiotemporal features, human activity recognition.

Resumen

El objetivo de este trabajo es proponer un esquema basado en metodologías kernel para apoyar el análisis de datos en video. La principal meta es revelar la información de mayor impacto contenida en datos de una alta dimensión y una alta correlación entre sí, esto con el fin de mejorar las representaciones de características y facilitar la etapa de entrenamiento. En el reconocimiento de actividad humana, los patrones relevantes albergan comportamientos no lineales y que se encuentran ocultos en los datos. Para exponer estos comportamientos, incluir información a priori acerca de relaciones ideales da indicios de las dependencias no lineales en los datos. En este trabajo se introducen tres metodologías para mejorar las representaciones de características y facilitar la etapa de entrenamiento. Primero, una representación basada en kernel se introduce para descubrir las relaciones mas importantes en un diccionario de códigos mediante la inclusión de un kernel que guarda relaciones ideales de los códigos. El segundo método propuesto es una herramienta para el análisis de la relevancia de características. Esta metodología encuentra el conjunto mas discriminativo de características espacio temporales para mejorar la separabilidad entre clases con respecto a un criterio de alineamiento de kernels centralizados. Finalmente, se propone un nuevo método para sintonizar el parámetro de C regularización en una SVM. Este método se basa en una metríca distancia entre dos kernels de similitudes. Los métodos propuestos se comparan con técnicas de lÃnea base en el estado del arte mostrando excelentes resultados en la tarea demejorar la representación de los datos y obtener unas altas tasas de acierto en la etapa de clasificación. **Palabras clave:** Bolsa de características (BoF), CKA, representaciones Kernel, SVMs, características espacio temporales, reconocimiento de actividades humanas.

Contents

List of Figures

List of Tables

List of Acronyms

BoF	Bag of Features
CKA	Centered Kernel Alignment
HOF	Histograms of Optical Flow
HOG	Histogram of Oriented Gradients
SIFT	Scale-Invariant Feature Transform
STIP	Space-Time Interest Points
SURF	Speeded-Up Robust Features
SVM	Support Vector Machine

Part I.

Preliminaries

1. Introduction

1.1. Motivation

Computer vision field has been widely studied in the last decades, in large part due to the fast spread of digital cameras and the quickly growing of the catalogs of multimedia data found online [23]. There are many applications in this field, some of these are of interest such as medical imaging, industrial processes control, video surveillance, human computer interfaces, among others. Particularly, video surveillance applications are nowadays a topic of research attention due to increased global security concerns and growing need for effective monitoring of public places such as shopping malls, airports among others. In this field is required detecting, tracking and recognition of interest objects, understanding and analyzing its dynamics and behaviors along the time into the video sequences, due to it is not feasible to analyze online by human supervision in the most of the cases [43, 68]. Some relevant video-based surveillance tasks include car traffic monitoring, pedestrian detection, people counting, human activity recognition, homeland security, and crowd behavior analysis.
A vast amount of information can be extracted from videos collected by surveillance cameras. However, a lot of this is usually not relevant and becomes a concern to obtain a suitable model for a video-surveillance task. Furthermore, the extracted information shows nonlinear behaviors that may be hidden in the input space. Kernel-based methods appear as a tool capable of analyzing and revealing the nonlinear relationships in data either through mapping the data from the original input space into a kernel feature space of higher dimensionality by a nonlinear function or solve a linear problem in the transformed kernel space. These methods allow the extraction of the most relevant features carrying information about the structure in the data. Thus, the kernel methods emerge as innovative techniques to facilitate the interpretability and analyzing of data in video surveillance applications [9, 53].

In a local context, research group *Signal Processing and Recognition Group* (SPRG) of the Universidad Nacional de Colombia has been working on the analysis of video data, in order to propose computer vision algorithms to support the development of automatic systems for video surveillance and video monitoring applications [33, 37]. Recently, the research group is interested in the analysis of human social behaviors and human action recognition to support to video surveillance systems. Besides, SPRG is also interested in motion analysis based on MoCAP system [17], sequences of depth images obtained by Kinect sensors [45], analysis of biosignals and biomedical images [4, 5, 60].

1.2. Problem statement

An automatic video surveillance system can be represented by a pipeline exposes in the Figure **1-1**, this is composed of six main stages: *(i)* video sequence acquisition, *(ii)* preprocessing data, *(iii)* feature estimation, *(iv)* relevance analysis, *(v)* training stage, *(vi)* High-level processing and/or making decision. Particularly, the feature estimation, relevance analysis and training stages are of great interest since these can be crucial in the final performance of the system.

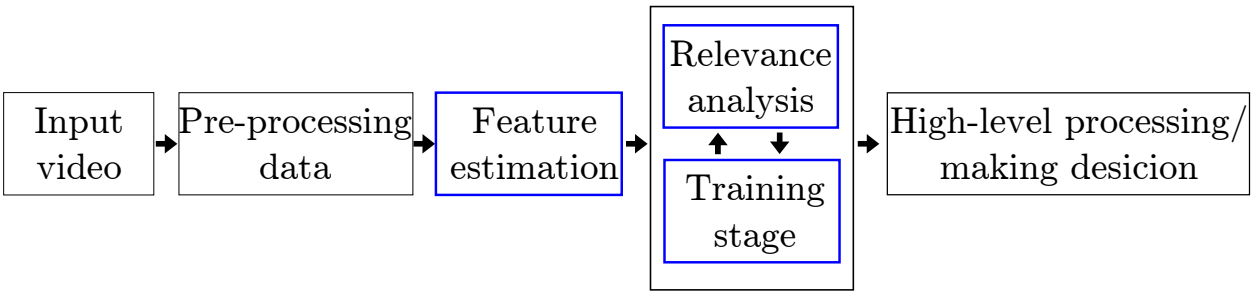


Figure 1-1.: General pipeline for computer vision system

From a video sequence, a lot of information is obtained, but not all of those are helpful for a particular video surveillance task. For this reason, the feature estimation stage plays a decisive role in choosing a suitable set of features. This fact can avoid future problems such as high dimensionality and high correlation in the data, which can add difficulty in developing a robust and an efficient video surveillance system. Besides, the features extracted must handle the well know problems in video applications such as posture, occlusion, illumination, scale, cluttered background, among others [15, 71, 74]. On the other hand, in video surveillance tasks there are large intra-class variations in individuals that belong to the same group or activity such as walking or running, principally caused by various motion speed changes and variations of view. In consequence, these carry a higher complexity to learn models of objects, people, activities, etc., regarding the training stage and the decision-making stage [41, 68]. For this reason, a relevance analysis can be necessary for revealing the better patterns in the data. Either to select a subset of useful features or a combination of these instead of using all of them in order to avoid the inclusion of noise variables in the training stage and easing the learning of a good model for human activity classification. In other words, the goal of relevance analysis stage is to select or extract a subset of features that increases the interclass and decreases the intra-class variability compared to any other subset of features, thus helping the training stage to learn a good model of classification and improve the final system performance [19, 36, 46].

From the training stage point of view, the problem of unbalanced classes is another challenge in video-based surveillance applications. It means most of the samples correspond to the usual activities such as walking or running, while, only a small percentage are infrequently

activities such as a person leaving a bag on a street. Regarding this, the learned model for classifying these activities will be biased for the activities with a major number of training samples generating high error rates on the others with a few training samples [2, 65]. To that end, to obtain a good performance in the human activity classification, properly setting the classifier parameters is crucial. Include information about how much unbalance that present in the human activity classes and similarity relationships among them can give salient cues about patterns in the data which may help to set these parameters obtaining a better discrimination among the classes. Also, this highlights the most significant samples to learn a generalist model of activity classification improving both the accuracy and decision-making stage.
In short, concerning the video surveillance task, choose a set of features that model the human behaviors more adequately is decisive for getting a good performance of the system. Also, reveal the most significant patterns in the data by an analysis of relevance help to avoid the inclusion of noise variables in the training stage. Finally, the unbalanced classes problem must be considered for setting the classifier parameters in the training stage. Include information of the amount of unbalance in the classes highlights the most relevant samples to get a good performance of the classification model learned.

1.3. Literature review

Commonly, computer vision applications can be represented by six main stages such as shown in the Figure **1-1**, which are: video sequence acquisition, pre-processing, feature estimation, relevance analysis, training stage, and high-level processing/decision making. Particularly, feature estimation stage is the first challenge to design a video-based system. The most promising current approaches are primarily based on low and mid-level features such as local spatiotemporal features. These one have drawn attention for content-based video analysis and specifically for human activity recognition [48]. The features based on Histograms of Oriented Gradients (HOG) describe the local appearance and shape of an object by a distribution of intensities of gradients or direction of contours [12, 13]. These methods operate on localized cells. Hence, this maintains the invariance to geometric and photometric transformations, but they are sensitive to material properties, textures, lighting and cluttered background [24, 69]. The features based on Histograms of Optical Flow (HOF) rely on the assumption that the differences in a sequence of images can be represented as a result of movement and velocities, rather that changes in material properties, illumination, among others [63, 69]. Most recent works have demonstrated that local spatiotemporal features are more robust to posture, occlusion, illumination, and cluttered background [41, 74]. However, they have a high dependence on the extraction of a sufficient number of relevant interest points to describe a region of the image or current frame properly [44]. These representations describe the image region as a collection of independent patches. These local features are typically divided into two parts: a detector and a descriptor. First, the detector localizes interest

points in a spatiotemporal space while the descriptor computes representations around the detected points. Finally, the patches are combined into a final representation.

Many feature detectors have been developed [15, 62, 70], and descriptors [26, 29, 30, 62]. Among these detectors and descriptors of local features, Scale-Invariant Feature Transform (SIFT) [34], Space-Time Interest Points (STIPs) [28] and Speeded-Up Robust Features (SURF) [6] are widely used due their ease of use and good performance. They are capable of catching spatial and temporal local dynamics in the image. It covers the simple and short activities such as primitive and periodic actions like walking, running, jumping, waving, among others in controlled scenes which of videos are taken by fixed cameras with uniform backgrounds, minor changes in lighting and small occlusions [38, 61]. For activities with more complex dynamics than the primitives, it is necessary analyzing sequences of local features instead of features by themselves for detecting human activities. These sequences or trajectories of features track a given interest point on the image along the time, this allows capturing the motion information in addition to the local information encoded by the descriptor at each location in space and time around the point. The key here is to discover relationships between a set of trajectories and a determinate class or activity to discriminate each class of the others and alleviate the classification stage, in other words, find the set of common trajectories present in each activity [56, 62, 63].

In the most video surveillance applications, an automated recognition of high-level activities is required. These involve recognition of human behaviors such as 'a person manifesting interest in something in a supermarket', detection of suspicious activities such as 'a person leaving a bag on the street' or interactions that involve two o more persons and/or objects like 'a group of people marching' or 'having a meeting'. Besides, high-level activities can be composed of multiple single actions and trajectories [2]. To encode these set of multiple actions, approaches based on Bag of Features (BoF) with spatiotemporal features have dominated the research work for human activity recognition [48, 56, 67, 75].

The BoF approaches use a pre-built dictionary that keeps the codes to quantize the occurrence of patterns in trajectories or regions of the image, and through of a combination of them be able to describe each activity, like classifying of a text for its words. The BoF approaches exhibit a high dimensionality by cause of the number of words that dictionary contains, so this increases the computational cost and complexity of data representation. In addition, these approaches expose high correlations among the classes and large intra-class variations what makes more challenging to set the classifier parameters in the training stage. To reduce the high dimension in the data, many authors use Principal Component Analysis (PCA). This approach is based on the properties and distribution of the data to determinate a new feature space of linearly uncorrelated variables, which the data are efficiently represented regarding independence between their dimensions. The number of used principal components is less than the original variables. Yacoob and Black [73] have considered a model in which recognition of various actions was treated by a different type of features (set of eigenvectors). They used PCA-based modeling and singular value decomposition technique

to represent activities. Lu and Little [35] apply PCA after calculating the HOG descriptor, which significantly reduce the dimension. Wang et al. [67], to improve the process efficiency, PCA is select the components whose contribute rate is bigger than a threshold. Xian et al. [71] applied PCA for the dimensionality reduction by half over STIP-HOG/HOF.

Also, PCA is used to select relevant features or find a new space of representation by linear combinations of the original variables to handle the high correlations and minimize the intra-class variance. Other works have proposed a search for combinations the original variables to choose the most relevant features according to the classification performance. Hall [20] introduce a correlation-based feature selection to ranks features subsets according to the correlation based on the heuristic of merit. John et al. [22] propose wrapper, this method uses a search algorithm to go through the whole combination of features. Ribeiro et al. [46] use an exhaustive search on the overall set of features, they consider all possible combinations. Pereira et al. [42] use gain information and relief-F over a multi-scale histogram of features to evaluate the importance of each bin . These approaches seek the best combination of a set of features, as a consequence, the execution time for obtaining the desired results could be huge.

Many times, the relationships in the data and its dynamics keep non-linear behaviors. Therefore, the relevant patterns can not be revealed by linear relations or linear combinations of the original variables [36, 58]. Many methods have been proposed to handle nonlinear relations in the video data. Kernel PCA performs traditional PCA in a so-called kernel feature space, which is nonlinearly related to the input space [21, 49, 56, 59]. Wang and Suter [66] extract features from the sequence of silhouettes by non-linear dimensionality reduction with kernel PCA. Noguchi and Yanai [38] propose a multiple feature fusion by Multiple Kernel Learning (MKL) for feature-fusion-based to select several useful features of trajectories. Lui et al. [31] proposed a model to feature selection called Markov Semantic Model to improve the stage of classification based on nonlinear SVMs. Xu et al. [72] propose a novel temporal graph kernel to measure the similarity between two string representation to capture spatial structures as well as high order and their nonlinear relationships. Although these approaches seek a better representation of the data to expose nonlinear patterns, they do not consider prior information about the classes. This information can weigh the importance of each feature for a particular class.

Most methodologies use Support Vector Machines (SVMs) at the stage of decision or classification due to its excellent performance to discriminate among the actions [48, 51, 56, 76]. But to achieve good performance of SVMs is necessary tuning its parameters properly. Many methodologies optimize the SVMs parameters by an exhaustive search over a grid of values for each parameter and choose the settings with the best performance in a validation stage [62, 63, 71, 74]. These methods do not take into account the existing overlap between classes, the number of samples and the correlations between them. Generally, SVMs include slack variables to handle with overlapping. These variables allow data points on the incorrect side of the margin boundary have a penalty that increases with the distance from it [25, 49, 55].

Sykens et al. [54] propose an error model to penalize overlapped samples in an LS-SVM. The approach assigns high weights to samples near the margin boundary and null weights too far away samples.Besides, The task of identifying activity classes is very challenging because of the imbalanced nature of such dataset where some events occur more frequently than others [39]. Shao et al. propose WLTSVM for imbalanced data classification [52]. They introduce a graph-based under-sampling strategy, which is robustness to outliers data points. In this sense, in the training stage is necessary to take into consideration the information about the imbalanced showed by some activity classes, and include this one helping the tuning of classifier parameters a smart way in order to improve de final performance of the system.

1.4. Objectives

1.4.1. General objective

To develop a kernel representation framework capable to support the video data analysis for video surveillance applications. This framework must be useful for disclosing the nonlinear salient patterns from available input data to enhance the feature representations, and support the classification stage to achieve a good performance in terms of classification rates and data interpretability.

1.4.2. Specific objectives

- To elaborate a Kernel-based representation for improving spatiotemporal and trajectory descriptors based on Bag of Features (BoF) for modeling simple and complex human behavior dynamics favoring interclass separability in terms of supervised classification metrics.
- To develop a Kernel-based Relevance analysis framework in video data. The proposed framework must allow determining the salient features to describe a human activity class by including prior knowledge. The framework is tested in terms of supervised classification metrics and data interpretability.
- To develop an automatic methodology for tuning the C parameter in an SVM classifier by similarity relationships in the data based on kernel representations. The proposal seeks to penalize the samples with little similarities regarding their class allowing misclassification of these samples for improving the SVM model. The methodology performance must be tested in terms of the supervised classification metrics.

Part II.

Mathematical preliminaries

2. Mathematical preliminaries

2.1. Reproducing kernel Hilbert spaces

Let $\mathscr{X}$ be a set and $\mathscr{F}$ be a vector space of functions from $\mathscr{X}$ to the field $\mathbb{F}$; in particular, let $\mathbb{F}$=$\mathbb{R}$. Then, there exits a reproducing kernel Hilbert space (RKHS) $\mathscr{H}$ on $\mathscr{X}$ over $\mathbb{R}$, if:

- $\mathscr{H}$ is a vector subspace of $\mathscr{F}$.
- $\mathscr{H}$ is endowed with an inner product, $\langle\cdot,\cdot\rangle_{\mathscr{H}}$, and is complete in the metric induced by it.
- For every $x\in\mathscr{X}$ and $f\in\mathscr{H}$, the linear evaluation functional $F_x : \mathscr{H} \to \mathbb{R}$, defined as $F_x(f) = f(x)$, is bounded.

From the Riez theorem [27], it is known that for any bounded functional H on a Hilbert space $\mathscr{H}$, there exists a unique vector $h\in\mathscr{H}$ such that: $H(f)=\langle h, f\rangle_{\mathscr{H}}$ for all $f\in\mathscr{H}$. In turn, for each evaluation functionals F_x there exist a corresponding vector $\kappa_x\in\mathscr{H}$. The bivariate function defined by:

$$\kappa(x, x') = \kappa_x(x') \tag{2-1}$$

is called a reproducing kernel for $\mathscr{H}$, with $x'\in\mathscr{X}$. So, it can be verified that

$$\kappa(x, x') = \langle\kappa_x, \kappa_{x'}\rangle_{\mathscr{H}} \tag{2-2}$$

and $\|F_x\|^2_{\mathscr{H}}=\|\kappa_x\|^2_{\mathscr{H}} = \langle\kappa_x, \kappa_{x'}\rangle_{\mathscr{H}}=\kappa(x, x)$, where $\|\cdot\|$ stands for the norm operator.
Let $\mathscr{H}$ be a RKHS on the set $\mathscr{X}$ with kernel κ. The linear span of $\{\kappa(x,\cdot) : x\in\mathscr{X}\}$ is dense in $\mathscr{H}$. This results from the fact that any function f orthogonal to the span of $\{\kappa(x,\cdot) : x\in\mathscr{X}\}$ must satisfy $\langle f, \kappa_x\rangle_{\mathscr{H}}$, and thus $f(x)$=0.

Lemma 2.1.1. *Let $\{f_n\} \subset \mathscr{H}$, being $n\in\mathbb{N}$ an index counter. If $\lim_{n\to+\infty} \|f_n - f\|_{\mathscr{H}} = 0$, then $f(x)=\lim_{n\to+\infty} f_n(x)$ for every $x\in\mathscr{X}$.*

Proof 2.1.1. *This is a simple consequence of the reproducing property and Cauchy-Schwarz inequality:*

$$|f_n(x) - f(x)| = |\langle f_n - f, \kappa_x\rangle_{\mathscr{H}}| \leq \|f_n - f\|_{\mathscr{H}} \|\kappa_x\|_{\mathscr{H}} \to 0$$

□

Proposition 2.1.1. *Let $\mathscr{H}_1$ and $\mathscr{H}_2$ be RKHS on $\mathscr{X}$ with kernels κ_1 and κ_2, respectively. If $\kappa_1(x,x')=\kappa_2(x,x')$ for all $x,x'\in\mathscr{X}$, then $\mathscr{H}_1=\mathscr{H}_2$ and $\|f\|_{\mathscr{H}_1}=\|f\|_{\mathscr{H}_2}$ for every f.*

Proof 2.1.2. *we can take $\kappa(x,x')=\kappa_1(x,x')=\kappa_2(x,x')$ and thus the $M_l=\text{span}\{\kappa_x\in M_l : x\in\mathscr{X}\}$ is dense in $\mathscr{H}_l$, and for any $f(x)=\sum_n \alpha_n\kappa_{x_n}(x)$ there is no regard about whether f belongs to either M_1 or M_2. Note that $\|f\|^2_{\mathscr{H}_1}=\sum_{n,n'}\alpha_n\alpha_{n'}\kappa(x_n,x_{n'})=\|f\|^2_{\mathscr{H}_2}$, and thus $\|f\|_{\mathscr{H}_1}=\|f\|_{\mathscr{H}_2}$ for every $f\in M_1=M_2$. If $f\in\mathscr{H}_1$, then there is a sequence of functions $\{f_n\}\subset M_1$ that converge to f in norm. Since $\{f_n\}$ is Cauchy in M_1 is also Cauchy in M_2, so by completeness of $\mathscr{H}_2$ there exist $g\in\mathscr{H}_2$ such that $f_n\to g$. Then, by Lemma 2.1.1 we have that $f(x)=\lim_{n\to+\infty} f_n(x)=g(x)$ for every $x\in\mathscr{X}$, thus every $f\in\mathscr{H}_1$ is also in $\mathscr{H}_2$ and vice versa, and $\mathscr{H}_1=\mathscr{H}_2$. Finally, we can extend $\|f\|_{\mathscr{H}_1}=\|f\|_{\mathscr{H}_2}$ to all $\mathscr{H}_1$ and $\mathscr{H}_2$.*

□

Thus, two different RKHSs do not have the same reproducing kernel. The following theorem shows an alternative way to express the reproducing kernel of a RKHS $\mathscr{H}$.

Theorem 2.1.1. *Let $\mathscr{H}$ have reproducing kernel κ. if $\{e_\lambda : \lambda\in\Lambda\}$ is an orthonormal basis of $\mathscr{H}$, then:*

$$\kappa(x,x') = \sum_{\lambda\in\Lambda} e_\lambda(x)e_\lambda(x'), \tag{2-3}$$

where the series converges point-wise.

Proof 2.1.3. *For a fixed $\{x_n\}\subseteq\mathscr{X}$, we have:*

$$\sum_{n,n'=1}^{N}\alpha_n\alpha_{n'}\kappa(x_n,x_{n'}) = \left\langle \sum_{n=1}^{N}\alpha_n\kappa_{x_n}, \sum_{n'=1}^{N}\alpha_{n'}\kappa_{x_{n'}} \right\rangle_{\mathscr{H}} = \left\| \sum_{n=1}^{N}\alpha_n\kappa_{x_n} \right\|_{\mathscr{H}} \geq 0$$

□

Added to that, the Moore's Theorem is introduced, which is the converse to the above result and provides us a characterization of a positive definite function to be a sufficient condition for the function to be the reproducing kernel of some RKHS.

Theorem 2.1.2. *Let $\mathscr{X}$ be a set and $\kappa:\mathscr{X}\times\mathscr{X}\to\mathbb{R}$ be a positive definite function. Then, there exits a RKHS $\mathscr{H}$ of functions on $\mathscr{X}$, such that, κ is the reproducing kernel of $\mathscr{H}$.*

Proof 2.1.4. *Consider the functions $\kappa_x(x')=\kappa(x,x')$ and the space W spanned by the set $\{\kappa_x : x\in\mathscr{X}\}$. The following bilinear map $B:W\times W\to\mathbb{R}$:*

$$B\left(\sum_i \alpha_n\kappa_{x_n}, \sum_{n'}\beta_{n'}\kappa_{x_{n'}}\right) = \sum_{n,n'}\alpha_n\beta_{n'}\kappa(x_n,x_{n'}),$$

where $\alpha_n\beta_{n'}\in\mathbb{R}$, is well defined on W. To support the above claim, notice that if $f(x)=\sum_n \alpha_n\kappa_{x_n}(x)$ is zero for all $x\in\mathscr{X}$, then by definition $B(f,\kappa_x)=0$ for all x. Conversely, if $B(f,w)=0$ for all $w\in W$, then by taking $w=\kappa_x$ it can be seen that $f(x)=0$. Then, B is well defined.
Since κ is positive definite $B(f,f)\geq 0$ and we see that $B(f,f)=0$ if and only if $B(w,f)=0$ for all $w\in W$, therefore $f(x)=0$ for all $\mathscr{X}$. Now we have shown that W is a pre-Hilbert space with inner product B. Let $\mathscr{H}$ denote the completion of W, we need to show that every element of $\mathscr{H}$ is function on $\mathscr{X}$. Let $h\in\mathscr{H}$ be the limit point of a Cauchy sequence $\{f_n\}\subseteq W$. By Cauchy-Schwarz inequality:

$$|f_n(x)-f_{n'}(x)| = |B(f_n-f_{n'},\kappa_x)| \leq \|f_n-f_{n'}\|\kappa(x,x).$$

Therefore, the point-wise limit $h(x)=\lim_{n\to+\infty} f_n(x)$ is well defined. Concluding, let $\langle\cdot,\cdot\rangle_{\mathscr{H}}$ be the inner product on $\mathscr{H}$. Then, we have $\langle h,\kappa_x\rangle_{\mathscr{H}}=\lim_{n\to+\infty}\langle f_n,\kappa_x\rangle_{\mathscr{H}}=\lim_{n\to+\infty} B(f_n,\kappa_x)=h(x)$. Thus $\mathscr{H}$ is a RHKS with reproducing kernel κ.

□

Combining Proposition 2.1.1 with the Moore's Theorem (Theorem 2.1.2) shows the correspondence between RKHS's on the set $\mathscr{X}$ and positive definite functions on this set.

2.2. The covariance function

Consider a stochastic process $\{X(t):t\in\tau\}$, where $X(t)$ are real random variables defined on a probability space $(\Omega,\mathscr{B},\mathscr{P})$ with bounded second order moments, that is:

$$\mathbb{E}_t\left\{|X(t)|^2\right\} = \int_\Omega |X(t)|^2 d\mathscr{P} < \infty, \tag{2-4}$$

where $\mathbb{E}\{\cdot\}$ stands for the expectation operator. Without loss of generality, we can consider random variables with zero mean, $\mathbb{E}_t\{X(t)\}=0$ for all $t\in\tau$. The covariance function is defined as:

$$R(t,t') = \mathbb{E}_{t,t'}\{X(t)X(t')\} = \int_\Omega X(t)X(t')d\mathscr{P}, \tag{2-5}$$

where $t,t'\in\tau$. It is easy to verify that $R:\tau\times\tau\to\mathbb{R}$ is a positive definite function and therefore defines a RKHS of functions on τ. A result originally due to Loeve and presented by Parzen in [40] showed a congruence map between the RKHS induced by the function R on L_2 space that corresponds to the completion of the span of the set $\{X(t):t\in\tau\}$ denoted by $L_2(X(t):t\in\tau)$.

Theorem 2.2.1. *Let $\{X(t):t\in\tau\}$ be a random process with covariance kernel R. Then $L_2(X(t):t\in\tau)$ is congruent with the RKHS $\mathscr{H}$ with reproducing kernel R. Furthermore, any linear map $\phi_R:\mathscr{H}\to L_2(X(t))$ which has the property that for any $f\in\mathscr{H}$ and any $t\in\tau$*

$$\mathbb{E}_t \{\phi_R(f) X(t)\} = f(t) \tag{2-6}$$

is the congruence from $\mathscr{H}$ *onto* $L_2(X(t))$, *which maps* $R(t,\cdot)$ *into* $X(t)$.

2.3. Reproducing kernel Hilbert spaces in machine learning

It is universally acknowledged that the study of positive definite kernels is a topic of interest for the machine learning community as a generalization of a well body of theory that has been developed for linear models. In this way, a positive definite kernel κ is an implicit way to represent the samples of the input space $\mathscr{X}$. Owing to there is a correspondence between κ and a RKHS of functions $\mathscr{H}$, the kernel can be understood as an indirect way to compute inner products between elements of a Hilbert space that are the result of mapping the elements of $\mathscr{X}$ to $\mathscr{H}$. So, there is a mapping function $\varphi : \mathscr{X} \to \mathscr{H}$ such that:

$$\kappa(x, x') = \langle \varphi(x), \varphi(x') \rangle_{\mathscr{H}}. \tag{2-7}$$

Regarding this, the space $\mathscr{H}$ can be viewed as a feature space and φ is called the feature map. Consequently, by performing linear operations in $\mathscr{H}$ it is possible to perform nonlinear manipulations in the input space $\mathscr{X}$, however, there is no need to perform any explicit computations in $\mathscr{H}$ (see Figure 2-1).

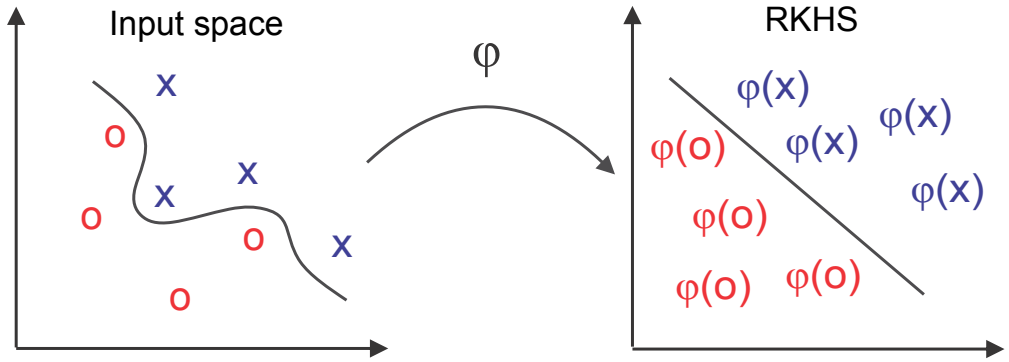


Figure 2-1.: kernel-based mapping.

Note that this idea is completely different to the congruence map introduced in Theorem 2.2.1. Then, an important property associated with the use of positive definite kernels in machine learning is the so-called representer theorem[50]:

Theorem 2.3.1. *Let* $\Omega : [0, +\infty) \to \mathbb{R}$ *be a strictly monotonic increasing function,* $\mathscr{X}$ *be a set, and* $\epsilon : (\mathscr{X} \times \mathbb{R}^2)^N \to \mathbb{R} \cup \infty$ *be an arbitrary loss function. Then, each minimizer* $f \in \mathscr{H}$ *of the regularized risk functional:*

$$\epsilon\left((x_1, y_1, f(x_1)), \ldots, (x_N, y_N, f(x_N))\right) + \Omega\left(\|f\|^2_{\mathscr{H}}\right), \tag{2-8}$$

admits a representation of the form:

$$f(x) = \sum_{n=1}^{N} \alpha_n \kappa(x_n, x), \tag{2-9}$$

where each $y_n \in \mathbb{R}$ *is a given output associated with the input* $x_n \in \mathscr{X}$.

Proof 2.3.1. *Let* $S=\text{span}\{\kappa(x_n, \cdot) : x_n \in \mathscr{X}, n \in [1, N]\}$ *denotes the subspace of* $\mathscr{H}$ *spanned by the* N *training samples. Consider the solution* $f \in \mathscr{H}$, *this solution can be written as:* $f = f_S + f_{S^\perp}$, *where* $f_S \in S$, $f_{S^\perp} \in S^\perp$, *and* $\perp$ *stands for the orthogonal symbol. Consequently,* $f(x_n) = f_S(x_n) + f_{S^\perp}(x_n) = f_S(x_n) + 0$. *Now, for the second term of the regularized risk funtional:*

$$\Omega\left(\|f\|^2_{\mathscr{H}}\right) = \Omega\left(\|f_S\|^2_{\mathscr{H}} + \|f_{S^\perp}\|^2_{\mathscr{H}}\right),$$

since Ω *is strictly monotonic increasing it is possible to see that the minimum will be achieved for* $\|f_{S^\perp}\| = 0$, *which implies that* $f_{S^\perp} = 0$.

□

With this in mind, it is possible to conclude that the representer theorem basically states that the solution of the minimization of the regularized risk functional can be expressed in term of the so-called training sample $\{(x_n, y_n) : n \in [1, N]\}$. Therefore, it allows us to deal with problems that a first glance appear to be infinite dimensional. Nonetheless, the regularization does not prevent of having local multiple minima, such a property requires some extra conditions, namely, convexity.

Part III.

Materials and Methods

3. Improving Spatiotemporal features and trajectories for human activity recognition based on Kernel Representation

The spatiotemporal features and trajectories of features approaches have had success in the human activity recognition task. The first method detects key points over a region, and then, a descriptor or sundry of them catch the spatial and temporal information around each key-point. All points with their descriptors are considered individually to represent an activity class [44, 57, 64]. The second, it tracks the key-points along the frames capturing both spatiotemporal and movement information. This method allows analyzing human activities with more complex dynamics than simple actions supported by the first method [48, 62]. Both methods combined with BoF approaches achieve good results in the task [41]. We propose a Kernel-based methodology to improve the representation of pre-built codebook of BoF called CKA-BFR as illustrated in Figure **3-1**. We introduce prior information about ideal relationships between the activity classes and the labels of them to find a mapping space using Centered Kernel Alignment (CKA) methodology, where the intraclass relations are maximized to improve the classification performance.

3.1. Centered Kernel Alignment to improve BoF Representations (CKA-BFR)

Given a codebook $\boldsymbol{C} \in \mathbb{R}^{D \times P}$, with D codewords generated by running k-means with D centroids over the matrix of spatiotemporal features $\boldsymbol{X} \in \mathbb{R}^{N \times P}$ extracted of the training video sequence, where P represents the dimension of the key-point descriptors and the corresponding label vector $\boldsymbol{l}_C \in \mathbb{Z}^D$ of centroids per class into the codebook. CKA finds a mapping space where the most important patterns to describe the activity classes are disclosing by nonlinear combinations of the data. In order to know the initial relationships between codewords, we establish a similarity kernel $\boldsymbol{K}_C \in \mathbb{R}^{D \times D}$ from $\boldsymbol{C}$. We compute each pairwise relation, $k^C_{d,d'}$ among the codewords as:

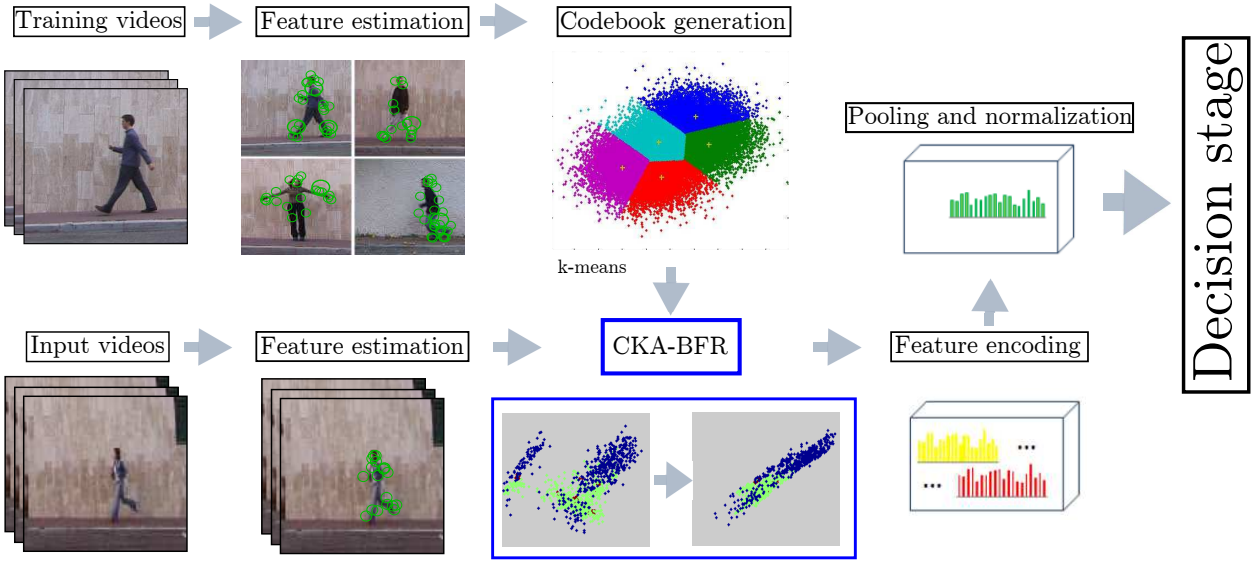


Figure 3-1.: CKA-BFR pipeline: The codebook is generated by running k-means over a set of features (keypoints or feature trajectories), and then, the CKA-BFR (blue boxes) maps the codebook to a new space where the relationships between the words maximize the separability of the activity classes and improving the descriptor representation of a new sample.

$$k^{C}_{d,d'} = \kappa_C\{\mathbf{d}_{CA}(C_d, C_{d'})\} :, d, d' \in \{1, \cdots, D\} \tag{3-1}$$

where $\mathbf{d}_{CA}(\cdot,\cdot) : \mathbb{R}^P \times \mathbb{R}^P \rightarrow \mathbb{R}$ is a distance operator implementing the positive definite kernel function $\kappa_C(\cdot)$. Here, we rely on the Mahalanobis distance defined in $P-$Dimensional space with inverse covariance matrix $\mathbf{A}\mathbf{A}^T$ as:

$$\mathbf{d}^2_A(C_d, C_{d'}) = (C_d - C_{d'})\mathbf{A}\mathbf{A}^T(C_d - C_{d'})^T \tag{3-2}$$

where matrix $\mathbf{A} \in \mathbb{R}^{P\times Q}$ holds the nonlinear projection of each codeword, $\mathbf{o}_d = C_d\mathbf{A}$, with $\mathbf{o}_d \in \mathbb{R}^Q$, $Q \leq P$. The former matrix $\boldsymbol{K}_C$ is a Gaussian kernel selected by its general approximating ability, it defined as below:

$$\kappa_C(C_d, C^{'}_d; \sigma) = \exp\left(\frac{-d^2(C_d, C^{'}_d)}{2\sigma^2}\right) \tag{3-3}$$

We propose learning the matrix $\mathbf{A}$ by including information about of ideal intraclass relationships given for $\boldsymbol{l}_C$ and link it with the initial ones kept by $\boldsymbol{K}_C$. The ideal information is embedding in the kernel matrix $\boldsymbol{K}_{l_C} \in \mathbb{R}^{D\times D}$. This kernel matrix holds the pairwise similarity between labels $\boldsymbol{l}_{Cd}, \boldsymbol{l}_{Cd'} \in \boldsymbol{l}_C\{d, d' \in [1, N]\}$. The former matrix holds elements $k^{l_C}_{dd'} = \kappa_{l_C}(\boldsymbol{l}_{Cd}, \boldsymbol{l}_{Cd'})$. Thus, the pairwise label similarity samples $\boldsymbol{l}_C, \boldsymbol{l}^{'}_C$ is as bellow:

$$\kappa_{l_C}(\boldsymbol{l}_C, \boldsymbol{l}'_C) = \pi_{l_C, l'_C} \tag{3-4}$$

Where π_{l_C, l'_C} is a delta function,

$$\pi_{l_C, l'_C} = \begin{cases} 1 & \text{if } \boldsymbol{l}_C = \boldsymbol{l}'_C \\ 0 & \text{Otherwise} \end{cases} \tag{3-5}$$

Hence, we force the initial relationships in $\boldsymbol{K}_C$, so that, they have a similar behavior to the ideal relationship in $\boldsymbol{K}_{l_C}$ taking advantage of CKA cost function as [8, 10]:

$$\rho(\bar{\boldsymbol{K}}_C, \bar{\boldsymbol{K}}_{l_C}) = \frac{\langle \bar{\boldsymbol{K}}_C, \bar{\boldsymbol{K}}_{l_C} \rangle_F}{\sqrt{\langle \bar{\boldsymbol{K}}_C, \bar{\boldsymbol{K}}_C \rangle_F \langle \bar{\boldsymbol{K}}_{l_C}, \bar{\boldsymbol{K}}_{l_C} \rangle_F}} \tag{3-6}$$

where $\langle \cdot, \cdot \rangle_F$ is the matrix-based Frobenius norm operator. Notation $\bar{\boldsymbol{K}}$ stands for the centered versions of the kernel matrix $\boldsymbol{K}$ that is calculated by using the matrix multiplication property as: $\bar{\boldsymbol{K}} = \bar{\boldsymbol{I}}\boldsymbol{K}\bar{\boldsymbol{I}}$, begin $\bar{\boldsymbol{I}} = \boldsymbol{I} - \mathbf{1}^T\mathbf{1}/N$ the empirical centering matrix, $\boldsymbol{I} \in \mathbb{R}^{D \times D}$ the identity matrix, and $\mathbf{1} \in \mathbb{R}^D$ the all-ones vector.
In this sense, the projection matrix $\boldsymbol{A}$ exposes linear combinations of original variables to reveal the relationships that minimize the intraclass variations and maximize the interclass differences. To compute the projection matrix $\boldsymbol{A}$ that parameterizes a Mahalanobis distances between pairwise of codewords, can be formulated based on CKA function as:

$$\boldsymbol{A}^* = \arg\max_{\boldsymbol{A}} \log(\rho(\bar{\boldsymbol{K}}_C, \bar{\boldsymbol{K}}_{l_C}; \boldsymbol{A})), \tag{3-7}$$

where the logarithm function is used for mathematical convenience. Once the projection matrix $\boldsymbol{A}^*$ is estimated, we calculate the mapped codebook in the CKA space $\boldsymbol{O} \in \mathbb{R}^{D \times Q}$ and the mapped feature matrix $\boldsymbol{Y} \in \mathbb{R}^{N \times Q}$ holding row vectors $\boldsymbol{y}_n$ to encode the linear combination of discriminative input features according to the prior knowledge of ideal realtionships considered in $\boldsymbol{K}_{\boldsymbol{l}}$ as bellow:

$$\boldsymbol{O} = \boldsymbol{C}\boldsymbol{A}^* \tag{3-8}$$

$$\boldsymbol{Y} = \boldsymbol{X}\boldsymbol{A}^* \tag{3-9}$$

3.2. CKA-BFR experiments

Our proposal to improve the BoF representations is tested on four databases (Section 3.2.1) to demonstrate its good performance in terms of well know supervised measures, recall (r), precision (p), harmonic mean of precision-recall (F1) and Accuracy (*Acc*) defined as

$$r = \frac{TP}{TP + FN} \tag{3-10}$$

$$p = \frac{TP}{TP + FP} \tag{3-11}$$

$$F1 = \frac{2 * p * r}{p + r} \tag{3-12}$$

$$Acc = \frac{TP + TN}{P + N} \tag{3-13}$$

Where TP is the true positives rate, FP is the false positive rate, FN is the false negative rate, P is the number of positive samples and N is the number of negative samples.
The proposal is compared with two methodologies. The first is the baseline based on PCA to reveal linear combinations in the data. Secondly, K-PCA is used to show nonlinear relationships in data. The results are indicated in Table **3-1** and Table **3-2** to spatiotemporal features and trajectories of features respectively.

3.2.1. Databases

- **KTH[1]:** Containing six types of periodic human actions (walking, jogging, running, boxing, hand waving and hand clappling) performance several times by 25 subjects in four different scenarios : outdoors s1, outdoors with scale variation s2, outdoors with different clothes s3 and indoors s4 as illustrated Figure 3.2(a). Currently the database contains 2391 sequences. All sequences were taken over homogeneous backgrounds with a static camera with 25fps frame rate. The sequences were downsampled to the spatial resolution of 160x120 pixels and have a length of four seconds in average [51].

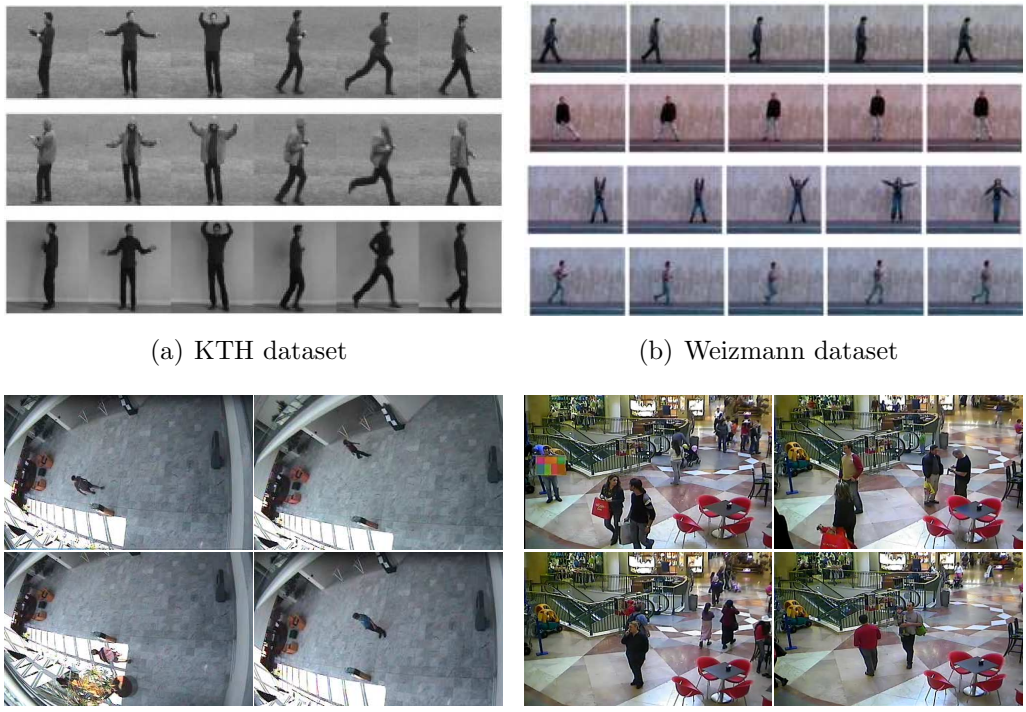
- **Weizmann[2]:** Containing ten types of periodic human actions(walk, run, jump, gallop sideways, bend, one-hand wave, two-hands wave, jump in place, jumping jack, skip) (See Figure 3.2(b)). The database contains 90 low-resolution(180×144), video sequence showing nine different people. All sequences were taken over homogeneous backgrounds with a static camera with 50 fps frame rate [7].

- **CAVIAR[3]:** Contains 26 video sequences of a corridor in a shopping center taken by a single camera with frame size of 384×288 and frame rate of 25fps. We use the social descriptors proposed on [42] which are Interested, Exploring and Distracted (See Figure 3.2(c)).

- **IIT-shopping:** Contains a video sequence in a shopping center taken by a single camera with frame resolution of 512×384 and frame rate of 25fps (See Figure 3.2(d)).

[1] http://www.nada.kth.se/cvap/actions/
[2] http://www.wisdom.weizmann.ac.il/~vision/SpaceTimeActions.html
[3] http://homepages.inf.ed.ac.uk/rbf/CAVIAR/

Besides, there is at least one pedestrian in 97,3 % of the frames and an average of 3,5 person per frame [1]. We use the social descriptors proposed on [42] which are Interested, Exploring and Distracted.



(a) KTH dataset (b) Weizmann dataset

(c) CAVIAR dataset (d) IIT dataset

Figure 3-2.: Datasets used to test CKA-BFR

3.2.2. Spatiotemporal features experiments

In this section, CKA-BFR is tested over spatiotemporal features by themselves to represent a human activity. First, the features are extracted from regions where there are persons manually labeled. These spatiotemporal features are obtained using the free ActionHOG library[4] proposed in [74]. The library extracts two types of low-level features including STIP-HOG/HOF and SURF/HMI-HOG. Each keypoint detected from STIP detector is described by HOG, HOF and HMI-HOG descriptors (128-dimensional vector by channel). The final representation of the keypoint is the 384-dimensional vector (128×3). Then, the codebook is generated by running k-means over the extracted features. It was run several times increasing the number of centroids from 100 to 8000 to determinate the optimal number of codewords. The performance is measured on the baseline methodology regarding F1 score average (Eq. 3-12) as illustrated in Figure **3-3**.

[4]https://github.com/xiaodongyang/ActionHOG

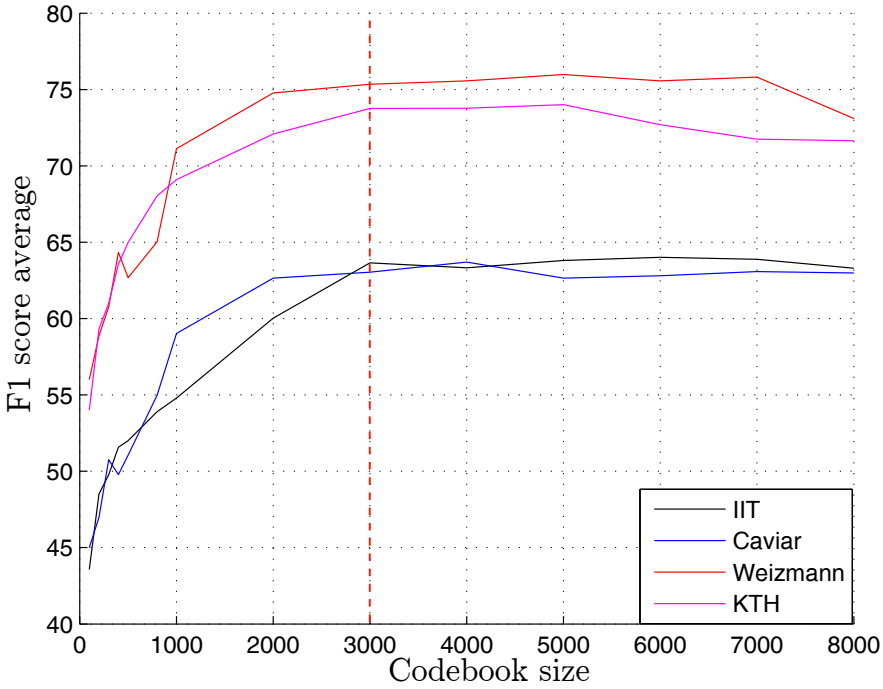


Figure 3-3.: Codebook size: k-means was running several times to estimate the codebook size regarding the F1 score average on the baseline methodology over spatio-temporal features. The codebook size is set to 3000 codewords for all databases.

A codebook of 3000 words is generated for each dataset, this is evaluated for the three approaches (PCA, K-PCA and CKA-BFR) to aim the best representation to improve the relationships in the codebook and also to improve the descriptor representation. The representation found by PCA with 90 % of variance retained reduces the codebook dimension from 384 to 235 same to CKA-BFR due to PCA gives the space dimension for this. K-PCA mapping the data from 384 to 212 dimensions. Then, $\boldsymbol{C}_{pca} \in \mathbb{R}^{3000\times 235}$, $\boldsymbol{C}_{kpca} \in \mathbb{R}^{3000\times 212}$ and $\boldsymbol{C}_{cka-bfr} \in \mathbb{R}^{3000\times 235}$. Finally, each sample is mapped to the three spaces by its matrix $\boldsymbol{A}$.

In the pooling and normalization stages, the mapped descriptors are assigned in a soft manner to the codebook. We use a gaussian kernel $G(\cdot,\cdot)$ to encode the similarity between the each mapped descriptor $\boldsymbol{y}_n$ and the codewords into $\boldsymbol{O}$ as

$$f_n(\boldsymbol{y}_n) = \frac{G(\boldsymbol{y}_n, \boldsymbol{o}_d; \sigma)}{\sum_{n=1}^{N} G(\boldsymbol{y}_n, \boldsymbol{o}_d; \sigma)} \tag{3-14}$$

Where $f_n \in \mathbb{R}^Q$ is the descriptor in a codebook representation. The σ parameter is the bandwidth of the Gaussian Kernel and it is calculated using Renyi's information metrics by the maximization of the information potential variability in the codebook $\boldsymbol{O}$ [3]. Then, the set of descriptors in a codebook representation that belongs a region of person (manually

obtained) are pooling and normalized as

$$\boldsymbol{F}_i = \frac{1}{M}\sum_{m=1}^{M} f_n(\boldsymbol{y}_m) \tag{3-15}$$

M is the number of descriptor in a specific region in a frame and $\boldsymbol{F} \in \mathbb{R}^{I\times Q}$, where I is the number of region of persons along the video sequence.
In the classification stages, we trained a knn classifier with 3, 5 and 7 neighbors following a hold-out validation with 70 % of data for training and 30 % for testing. We performed the experiment 30 times for each dataset with different data into partitions for training and testing. The results are shown in Table **3-1**.

Method	**Baseline (PCA)**				**K-PCA**				**CKA-BFR**			
Dataset	r	p	F1	Acc	r	p	F1	Acc	r	p	F1	Acc
KTH	77.78	72.83	73.11	90.94 ± 2.30	79.70	75.67	75.92	91.89 ± 2.45	**82.45**	**78.33**	**78.23**	**92.78 ± 2.28**
Weizmann	79.97	75.50	75.33	91.83 ± 2.10	82.29	79.50	79.63	93.17 ± 3.40	**84.42**	**82.00**	**82.12**	**94.00 ± 2.53**
Caviar	70.66	65.00	63.83	76.67 ± 3.01	71.05	65.67	65.24	77.11 ± 3.42	**73.72**	**71.67**	**71.56**	**81.11 ± 3.13**
ITT	71.31	66.00	63.74	77.33 ± 2.54	72.09	**69.67**	**67.84**	**79.78 ± 2.67**	**78.67**	69.33	64.90	79.56 ± 2.79

Table 3-1.: Classification results on all datasets using spatiotemporal features by themselves to describe a human activity

3.2.3. Feature trajectories experiments

Similar to the Section 3.2.2, CKA-BFR seeks a mapping matrix $\boldsymbol{A}$ to exposes nonlinear relationships between trajectories of features that maximize the class separability. The trajectories are computed as multi-scale histogram of features based on the Takahashi proposal [56]. The key-points along the trajectories are computed as:

$$P_u = [p_u^x, p_u^y] \tag{3-16}$$

$$p_u^x = [p_u^{x,t_1}, p_u^{x,t_1+1}, \ldots, p_u^{x,t_2}], \qquad p_u^y = [p_u^{y,t_1}, p_u^{y,t_1+1}, \ldots, p_u^{y,t_2}] \tag{3-17}$$

where P_u is a set of points in the T_u trajectory, p_u^x is the set of its x-coordinates, p_u^y is the set of its y-coordinates, and t_1 and t_2 are the starting and ending frames, respectively.
Each feature are encoded into a multi-scale histogram controlled by $R \in \mathbb{N}$, the number of granularity levels. Considering the feature f^0 extracted along the trajectory T_u, its multi-scale representation of size R is given by a $f_u = [H_u^1, H_u^2, \cdots, H_u^R]$, where each entry, H_u^r, is a normalized histogram of 2^{r+1} bins, for each $r = [1, 2, \cdots, R]$. The final representation for trajectory T_u is the concatenation of all the multi-scale feature histograms, and is given by a fixed-length vector

$$F_u = [(f_u^0), (f_u^1), \cdots, (f_u^{J-1})] \tag{3-18}$$

where J is the number of feature descriptors.
We used four relational descriptors based on [42] which are:

- *angular direction change* is the angular variation of movement of the individual between consecutive frames.
- *distance of interest* expresses the distance between individual positions and the object of interest.
- *direction of interest* wich is the manually gaze estimation direction of the head of the person.
- *velocity* expresses the instantaneous velocity of the individual at frame.

We compute the trajectories of features on the four datasets exposes in Section 3.2.1. For KTH and Weizmann, only the angular variation of movement is computed for each key-point detected in a region of person (manually labeled). We set $R = 4$ for the levels of granularity histogram bins. Namely, each key-point is described by four concatenated histograms of 4, 8, 16, and 32 bins (60-dimensional vector). For these datasets do not have information of the other relational descriptors. For Caviar and IIT-shopping datasets the information of the four relational descriptors are available with $R = 3$ levels of granularity histogram bins (4, 8 and 16)[5]. The final descriptor is 112-dimensional vector (28 bins for the descriptor histogram).
To set the codebook size k-means was running several times as illustrated in Figure **3-4**. Here, 1200 codewords are enough to obtain a good performance in a baseline methodology for Caviar and IIT-shopping dataset while KTH and Weizmann needed 1800 codewords to obtain a good performance on baseline. Then, the initial codebooks are $\boldsymbol{C}_{KTH} \in \mathbb{R}^{1800\times 60}$, $\boldsymbol{C}_{Weiz} \in \mathbb{R}^{1800\times 60}$, $\boldsymbol{C}_{Caviar} \in \mathbb{R}^{1200\times 112}$ and $\boldsymbol{C}_{IIT} \in \mathbb{R}^{1200\times 112}$
CKA-BFR computes the mapping matrix $\boldsymbol{A}$ to reveal nonlinear relationships in the trajectories of features as expose in Section 3.1 for the four codebooks. In the pooling and normalization stages, the set of descriptors in a region for KTH and Weizmann datasets are assigned to codebook representation by Equations (3-14) and (3-15), while the descriptors calculated in the Caviar and IIT datasets are assigned to its codebook representation by Equation (3-14) because a region is described by one trajectory instead of several such as in key-points trajectories.
In the classification stage, we trained a knn classifier with 3, 5 and 7 neighbors following a hold-out validation with 70 % of data for training and 30 % for testing. We executed the experiment 30 times for each dataset with different partitions of data for training and testing. The results are shown in Table **3-2**.

[5] We thank to Eduardo Marques [42] for sharing the information of the descriptors of the trajectories for Caviar and IIT-shopping datasets

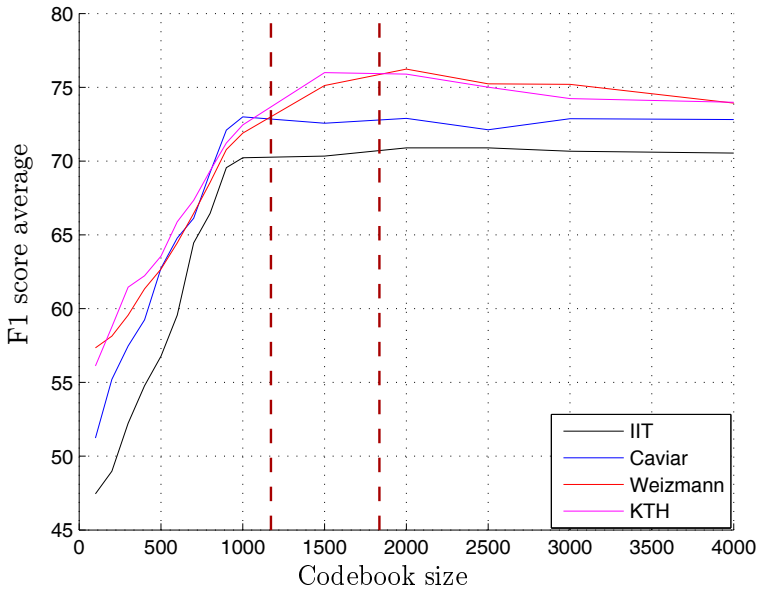


Figure 3-4.: Codebook size: k-means was running several times to estimate the codebook size regarding the F1 score average on baseline methodology over feature trajectories. The codebook size is set to 1200 codewords for Caviar and IIT and 1800 for KTH and Weizmann datasets.

Method	Baseline (PCA)				K-PCA				CKA-BFR			
Dataset	**r**	**p**	**F1**	**Acc**	**r**	**p**	**F1**	**Acc**	**r**	**p**	**F1**	**Acc**
KTH	80.43	75.67	76.08	91.89 ± 1.23	80.96	76.50	76.79	92.17 ± 2.01	**83.04**	**79.17**	**79.41**	**93.06 ± 1.78**
Weizmann	80.42	76.00	75.61	92.00 ± 2.10	83.58	81.00	81.15	93.67 ± 2.56	**85.39**	**83.17**	**83.30**	**94.39 ± 2.12**
Caviar	73.95	72.00	72.23	81.33 ± 1.05	75.00	72.03	71.84	81.40 ± 1.34	**76.78**	**76.33**	**76.31**	**84.22 ± 1.40**
ITT	76.43	71.33	70.68	80.89 ± 2.23	78.64	73.00	72.81	82.00± 1.98	**82.41**	**80.33**	**80.23**	**86.96 ± 1.51**

Table 3-2.: Classification results on datasets using trajectories of features to describe a human activity

3.3. Discussion

Figures 3-3 and 3-4 show the classification performance for all datasets. The spatiotemporal features are capable to achive a good performance regarding the F1 measure when periodic actions are evaluated, but these features are not enough to describe human actions with more complexity such as the activities into the Caviar and IIT datasets. The features based on trajectories improved the performance of periodic and complex activities in a baseline approach. Besides, it needs a shorter codebook (almost less of the half) than spatiotemporal features.

Table 3-1 shows the classification results on all datasets based on spatiotemporal features are used to describe a human activity. The baseline methods PCA and K-PCA exhibit a high accuracy rate for KTH and Weizmann datasets, but its recall and precision rate are not as good due to PCA discovers the most important patterns in classes with the most number of samples skewing the classifier model a some activity classes. Although K-PCA improves the both rates, CKA-BFR shows the best performance. Include prior information allows our methodology to reveal the most significant patterns per class, avoiding biased only the classes with more samples. For Caviar and IIT datasets, PCA and K-PCA have an acceptable performance considering that these datasets contain a high imbalance in some classes. CKA-BFR obtains the best rates of classification for Caviar while K-PCA gets the highest accuracy rate for IIT, but also a low recall rate because samples of the smaller classes are often misclassified (high rate of FN). CKA-BFR increases the recall rate (lower rate of FN) maintaining the accuarcy rate. In fact, CKA-BFR finds the best relationships between the codewords that maximize the separability of the activity classes.

In table 3-2 is shown the classification results of trajectories of features for describing a human activity. As we mentioned above, these improve the performance for all datasets. The trajectories have a lower correlation between them than spatiotemporal features. This issue is reflected in the fact that the codebook requires fewer codewords to discriminate between activity classes and in the best classification rates compared with Table 3-1.Besides, CKA-BFR improves the representation of the codebook achieved a high accuracy rate for all datasets with good precision and a good recall compared with the other methodologies.

4. Kernel-based feature Relevance Analysis for Video data information (KRAV)

Video-based human activity recognition is a very demanding task that extracts a very large set of features from diverse sources, such as motion, appearance, texture, among others. Such information normally presents high dimensionality. Therefore, it should undergo a discriminant analysis process to reduce and select the most prominent features. We propose a Kernel-based Relevance Analysis approach, called *KRAV*, which using two kernel functions takes advantage of the available joint information associating the employed features with the corresponding activity labels. Figure 4-1 illustrates the KRAV methodology. A set of social descriptors are computed over a region of a person. Then, a relevance analysis stage ranks the features by its importance, and a subset of these or an embedding are selected to describe in a better way each region and improve the stage of classification that assigns an individual profile (I.P) to each region (Disoriented, Distracted, Exploring, Interested).

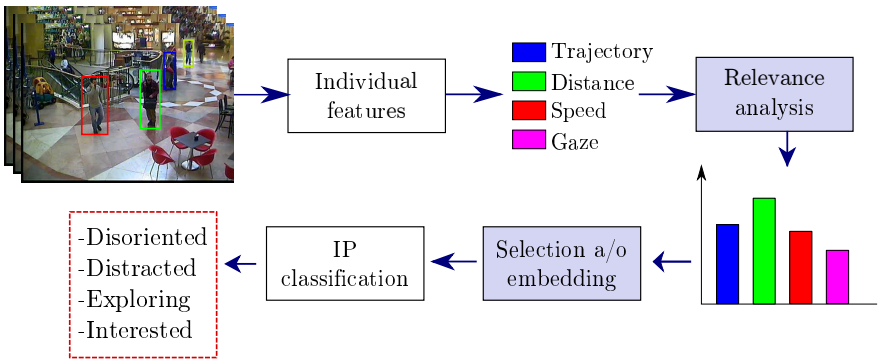


Figure 4-1.: KRAV pipeline: A set of features or a new embedding space is computed by relevance analysis based on Kernel representations

4.1. KRAV:Kernel-based feature Relevance Analysis for Video data

Given an input representation space $\boldsymbol{X}\in\mathbb{R}^{N\times P}$ with N samples and P spatio-temporal features and the corresponding social behaviour label vector $\boldsymbol{l}\in\mathbb{Z}^N$, we extract the following kernel matrices:

$$k^{\boldsymbol{X}}_{nn'} = \kappa_X\left(\boldsymbol{x}_n, \boldsymbol{x}'_n\right), \tag{4-1a}$$
$$k^{\boldsymbol{l}}_{nn'} = \kappa_L\left(l_n, l'_n\right), \tag{4-1b}$$

The former matrix, $\boldsymbol{K}_{\boldsymbol{X}}\in\mathbb{R}^{N\times N}$, holds the pairwise similarity between samples $\boldsymbol{x}_n, \boldsymbol{x}_{n'}\in\boldsymbol{X}$. Grounded on its general approximating ability and mathematical tractability, we select the Gaussian kernel defined as below:

$$\kappa_X\left(\boldsymbol{x}, \boldsymbol{x}'; \sigma\right) = \exp\left(-\,\mathrm{d}^2\left(\boldsymbol{x}, \boldsymbol{x}'\right)/2\sigma^2\right) \tag{4-2}$$

where $\mathrm{d}(\cdot,\cdot):\mathbb{R}^P\times\mathbb{R}^P\mapsto\mathbb{R}^+$ is a distance operator and $\sigma\in\mathbb{R}^+$ is the kernel bandwidth that rules the observation window within the similarity distance is assessed. We rely on the Mahalanobis distance to carry out the pairwise comparison between samples $\boldsymbol{x}_n$ and $\boldsymbol{x}_{n'}$ based on the Gaussian kernel κ_X. Therefore, the distance function in 4-2 is fixed as follows:

$$\mathrm{d}^2_{\boldsymbol{A}}(\boldsymbol{x}_n, \boldsymbol{x}_{n'}) = (\boldsymbol{x}_n - \boldsymbol{x}_{n'})\,\boldsymbol{A}\boldsymbol{A}^\top\,(\boldsymbol{x}_n - \boldsymbol{x}_{n'})^\top \tag{4-3}$$

where matrix $\boldsymbol{A}\in\mathbb{R}^{P\times M}$ holds the linear projection $\boldsymbol{y}_n=\boldsymbol{x}_n\boldsymbol{A}$, with $\boldsymbol{y}_n\in\mathbb{R}^M$, $M\leq P$, and $\boldsymbol{A}\boldsymbol{A}^\top$ is the corresponding inverse covariance matrix.
The latter kernel matrix, $\boldsymbol{K}_{\boldsymbol{l}}\in\mathbb{R}^{N\times N}$, holds the pairwise similarity between labels $l_n, l_{n'}\in\boldsymbol{l}$ $(n, n'\in[1, N])$. Namely, We set a positive definite kernel for the labels kernel κ_L that measures the pairwise similarity of $l, l'\in\mathcal{L}$ labels as below:

$$\kappa_L\left(l, l'\right) = \pi_{ll'} \tag{4-4}$$

where the delta function is $\pi_{ll'}=1$ if the samples have the same label $l=l'$, otherwise, $\pi_{ll'}=0$. To assess the joint information between the spatiotemporal features and the corresponding social behaviour labels, we evaluate how well the estimated kernel functions, $\kappa_{\boldsymbol{X}}$ and $\kappa_{\boldsymbol{L}}$, align each to other. To this end, the commonly-known centered kernel alignment (CKA) is applied that measures the similarity between a couple of characterizing kernel functions [18]. In particular, we employ the normalized inner product of both kernel functions to estimate the dependence between jointly sampled data as follows [8]:

$$\rho\left(\bar{\boldsymbol{K}}_{\boldsymbol{X}}, \bar{\boldsymbol{K}}_{\boldsymbol{l}}\right) = \frac{\langle\bar{\boldsymbol{K}}_{\boldsymbol{X}}, \bar{\boldsymbol{K}}_{\boldsymbol{l}}\rangle_{\mathrm{F}}}{\sqrt{\langle\bar{\boldsymbol{K}}_{\boldsymbol{X}}, \bar{\boldsymbol{K}}_{\boldsymbol{X}}\rangle_{\mathrm{F}}\langle\bar{\boldsymbol{K}}_{\boldsymbol{l}}, \bar{\boldsymbol{K}}_{\boldsymbol{l}}\rangle_{\mathrm{F}}}}, \tag{4-5}$$

where notation $\langle\cdot,\cdot\rangle_{\mathtt{F}}$ stands for the Frobenius matrix norm, $\bar{\boldsymbol{K}}$ is the centered version of the kernel matrix $\boldsymbol{K}$ calculated as $\bar{\boldsymbol{K}}{=}\tilde{\boldsymbol{I}}\boldsymbol{K}\tilde{\boldsymbol{I}}$, $\tilde{\boldsymbol{I}}{=}\boldsymbol{I}{-}\boldsymbol{1}^{\top}\boldsymbol{1}/N$ is the empirical centering matrix, $\boldsymbol{I}{\in}\mathbb{R}^{N\times N}$ is the identity matrix, and $\boldsymbol{1}{\in}\mathbb{R}^{N}$ is the all-ones vector.
It must be noted that for computation of the projection matrix $\boldsymbol{A}$ in 4-3, the CKA-based formulation, $\rho{\in}\mathbb{R}^{+}$, in 4-5 is integrated into the following kernel-based learner:

$$\widehat{\boldsymbol{A}} = \arg\max_{\boldsymbol{A}} \log\left(\rho\left(\bar{\boldsymbol{K}}_{\boldsymbol{X}}, \bar{\boldsymbol{K}}_{\boldsymbol{l}}; \boldsymbol{A}\right)\right), \tag{4-6}$$

where the logarithm function is used for mathematical convenience.
As a result, the estimated projection matrix $\widehat{\boldsymbol{A}}$ is differently affected by each input feature. Consequently, we quantify the contribution of each input feature for building $\widehat{\boldsymbol{A}}$ by introducing the relevance vector index, $\boldsymbol{\varrho}{\in}\mathbb{R}^{P}$ with elements computed as:

$$\varrho_p = \sum_{m=1}^{M} |a_{pm}|; \forall p{\in}P, \quad a_{pm} \in \widehat{\boldsymbol{A}} \tag{4-7}$$

The main rationale behind the proposed relevance index is that the features with bigger variability should favor a better kernel adjustment. Thus, the largest values of ϱ_p the better the input features. Furthermore, we rank the original feature set in terms of ϱ_p to carry out selection of the most discriminant spatiotemporal features. Thus, we choose $M_S{\leq}P$ features that have ϱ_p exceeding a certain threshold of relevance, resulting in $\widetilde{\boldsymbol{X}}{\in}\mathbb{R}^{N\times M_S}$.
Nonetheless, we further enhance the class inter-separability through the following embedding matrix: $\widetilde{\widetilde{\boldsymbol{X}}}{=}\widetilde{\boldsymbol{X}}\widetilde{\boldsymbol{A}}$, where $\widetilde{\widetilde{\boldsymbol{X}}}{\in}\mathbb{R}^{N\times M_E}$ and $\widetilde{\boldsymbol{A}}{\in}\mathbb{R}^{M_S\times M_E}$ is another rotation matrix computed from $\widetilde{\boldsymbol{X}}$ using 4-6, being $M_S{\geq}M_E$.
From now on, the above discussed approach of training is termed *Kernel-based Relevance Analysis for Video data* (*KRAV*).

4.2. KRAV experiments

The evaluation of *KRAV* is carried out by means of the following three tests: *1) Relevance vectors analysis:* Contrast and analyze the relevance vectors $\boldsymbol{\varrho}$ obtained by *KRAV* and state-of-the-art relevance analysis methods. *2) Finding the set of relevant features:* Calculate a classification performance curve for each method while adding the respective features selected as relevant. This to find a new space $\boldsymbol{X}_S$. *3) Embedding using the set of relevant features:* Use *KRAV* as a feature embedding tool to find new representation spaces from $\boldsymbol{X}_S$, looking for improving the classification performance and avoiding the calculation of unnecessary features. We tested KRAV over ITT - shopping dataset (3.2.1) using the four relational descriptors (angular direction change, distance of interest, gaze direction and velocity) based on [42] explained in Section 3.2.3.

4.2.1. KRAV setting

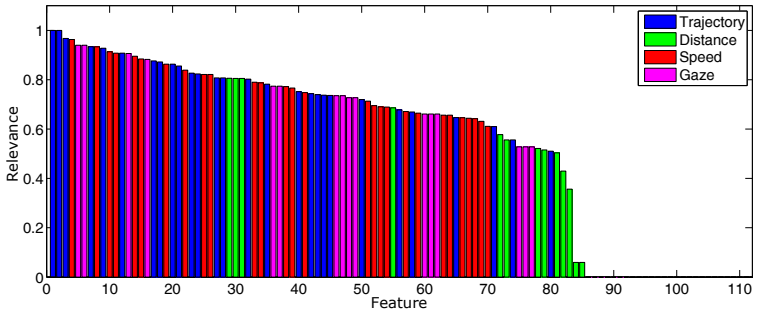
The proposed kernel-based relevance analysis method (see Section 4.1) to obtain the vector $\boldsymbol{\varrho}$ holding the most relevant features from the original space $\boldsymbol{X}$ is used. To this end, we adopted an iterative gradient descent algorithm for the optimization problem to find the projection matrix $\boldsymbol{A}$ shown in Eq. 4-6. Besides, to tune the Gaussian kernel bandwidth σ from Eq. 4-2, we use the information theoretic learning framework proposed in [3]. Given the high class unbalance exposed in ITT dataset (45 Distracted, 776 Exploring, 41 Interested and 7 disoriented subjects per class), we set the resulting values from function $\delta_{ll'}$ (Eq. 4-4) depending on the number of samples per class N_c as $\delta_{ll'}{=}1/N_c$ if $l{=}l'$, otherwise, $\delta_{ll'}{=}0$. As such, classes with few samples will be more relevant for the *KRAV* kernel-based learner, exposed in Eq.4-6. From matrix $\boldsymbol{X}$, we compute two new representation spaces $\boldsymbol{X}_s$ and $\boldsymbol{X}_e$, as explained in Section 4.1. Lastly, using $\boldsymbol{X}_s$ and $\boldsymbol{X}_e$, we perform the I.P. classification using a *k-nn* classifier with 10 folds and 10 iterations. The number of neighbours used for the classifier was heuristically found within the set $[1, 3, 5, 7, 9, 11]$. To deal with the unbalance of the *Diso.* class during the cross-validation, we randomly selected the 75 % of the samples for training and the remaining 25 % for testting in each iteration.

4.2.2. Relevance vectors analysis

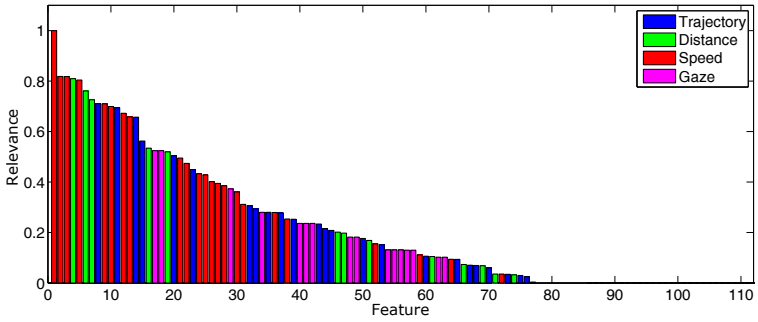
We first use *KRAV* as a feature selection tool to support the understanding of the salient aspects of the input feature set, facilitating the interpretation of the employed social behavior descriptors. To this, we calculate the relevance vector $\boldsymbol{\varrho}$ ranking the original feature set of $\boldsymbol{X}$ as explained in Section 4.2.1. For the sake of comparison, the proposed *KRAV* is compared with two baseline feature relevance methods. The first one is a variance-based relevance analysis (termed *VRA*) that ranks the input short-time features grounded on a variability criterion. Namely, *VRA* computes a relevance vector based on a linear transformation of the input representation space. Thus, *VRA* estimates the covariance between input features and the projection matrix maximizing the embedded space variability is fixed to computed such a linear transformation [14]. The percentage of retained variance parameter of *VRA* was set 90 %. The second baseline method called *Relief-f* calculates a relevance vector by looking the closest same and different class samples using a k-nn classifier [47]. The *Relief-f* parameter related to the number of nearest neighbors was set to 1.
The obtained relevance vector $\boldsymbol{\varrho}$ using *VRA*, *Relief-f* and the proposed *KRAV* for the individual descriptor is shown in Fig. **4-2**. Vectors $\boldsymbol{\varrho}$ are sorted in decreasing relevance order and normalized to the interval $[0, 1]$.

4.2.3. Finding the set of relevant features:

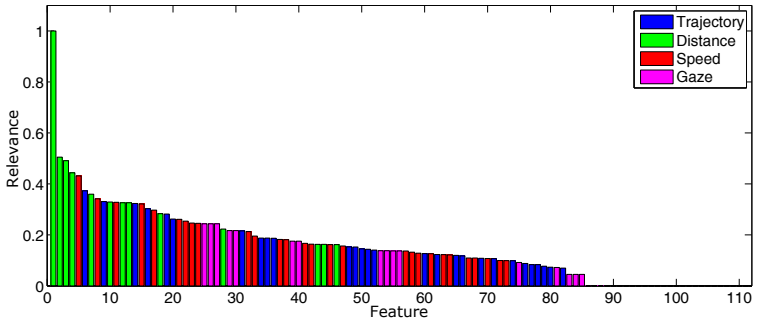
Then, to see the impact of the features selected as relevant for the I.P. classification, we calculate the performance curve through the k-nn cross-validation scheme explained in Sec-



(a) *VRA*



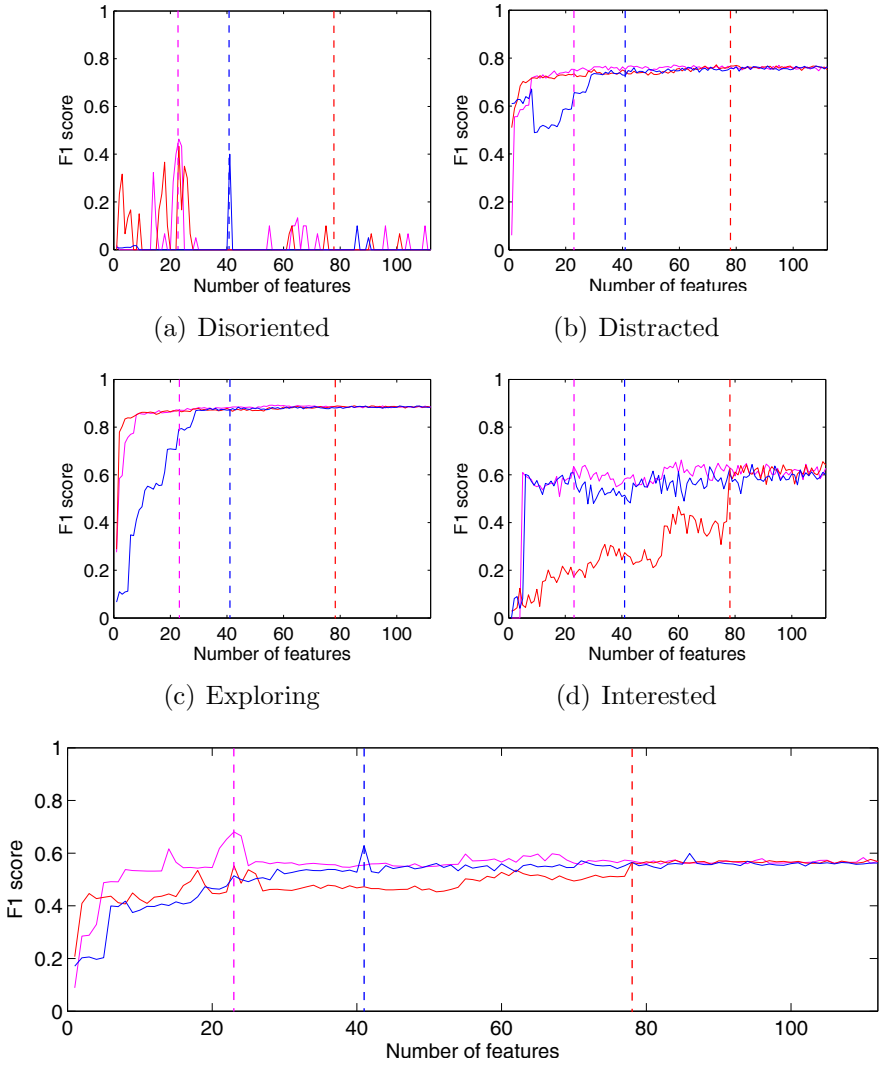
(b) *Relief-f*



(c) *KRAV*

Figure 4-2.: Feature ranking: the features are sorted in relevance to determinate its importance

tion 4.2.1, adding one by one the features ranked by the amplitude of the $\boldsymbol{\varrho}$ vector obtained for each method exposed in Fig. **4-2**. The classification performance for this experiment is assessed using the F_1 measure, which jointly consider the Precision and Recall. Fig. **4-3** shows the F_1 I.P. classification curve adding one by one the features ranked by the amplitude of $\boldsymbol{\varrho}$ for *VRA*, *Relief-f* and *KRAV*. The dashed lines indicate the selected subset of relevant features to conform $\boldsymbol{X}_S$ for each method, which corresponds to *VRA* $M_S = 78$, *Relief-f* $M_S = 41$ and *KRAV* $M_S = 23$. The threshold selection criteria to find M_S was set where the F_1 classification curve reaches the highest value.



(a) Disoriented (b) Distracted

(c) Exploring (d) Interested

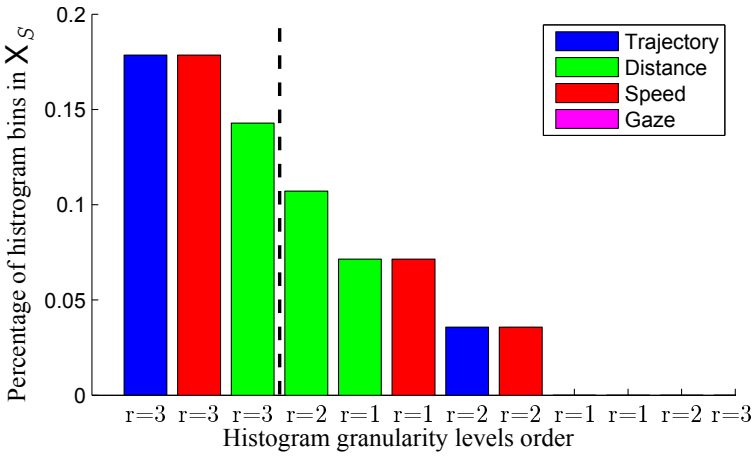
(e) Average

Figure 4-3.: IP classification results while adding relevant features. —*Relief-f*, —*VRA* and —*KRAV*. The dashed lines indicate the selected M_S for each method.

4.2.4. Embedding using the set of relevant features:

As a result of the above experiments, we obtain new feature spaces $\boldsymbol{X}_S$ using *KRAV* with the respective M_S for I.P. social behavior descriptors. With matrix $\boldsymbol{X}_S$, we perform two more tests using the proposed *KRAV* method. With these tests, we aim to see the impact of the non-linear embedding of the proposed method into the classification performance. Furthermore, we want to analyze if it is possible to avoid the calculation of some of the histogram features described in Section 3.2.3 maintaining a good classification. Following the different tests for *KRAV* are explained in detail:

- *KRAV-S*: Refers to *KRAV* as a feature selection, reporting the classification results for X_S exposed in Figure. 4.3(e).
- *KRAV-E*: We test *KRAV* as a feature embedding tool, which using the non-linear feature transformation explained in Section 4.1 generates a new representation space $\boldsymbol{X}_E$ from $\boldsymbol{X}_S$ with the goal of improving overall I.P. discrimination performance.
- *KRAV-R*: We use *KRAV* to perform a non-linear embedding using a feature subset of $\boldsymbol{X}_S$. Namely, we conform a new feature space $\boldsymbol{X}_R$ with the selected subset, which avoids the calculation of non-relevant histogram resolution levels for some features (see the I.P. descriptor explanation of Section 3.2.3) while keeping a good discrimination performance. Figure **4-4** shows the percentage of histogram bins per feature and resolution level in $\boldsymbol{X}_S$ for I.P.



(a) I.P. feature subset

Figure 4-4.: Feature subset obtained from $\boldsymbol{X}_S$ for I.P. descriptors

Table **4-1** shows the average F_1 measure by classes for the I.P. and G.B. classification using the aforementioned training scenarios, plus the results obtained by *VRA* and *Relief-f* reported in Section 4.2.3.

Method	I.P.	
	Avg F_1	#feat
VRA	0.5761	70 %
Relief-f	0.6288	37 %
KRAV-S	0.6807	21 %
KRAV-E	**0.7481**	21 %
KRAV-R	**0.7391**	12 %

Table 4-1.: $F1$ results and percentage of relevant characteristics for the IP classification using *VRA*, *Relief-f* and *KRAV* for feature selection and/or embedding.

4.3. Discussion

As seen, the relevance vectors obtained by each method highlight a different set of features as relevant for the I.P. classification. For the *VRA* method, most of the features provide similar information as shown in Fig. 4.2(a). Besides, the bins related to the Distance feature are not very relevant to classify the I.P.. The above can be explained due to *VRA* finds a linear combination of features which maximizes the variability among data samples. However, in this case, the variability criterion might not be the most proper choice to enhance the separability among classes since it does not include the supervised information of the labels. In contrast, the *Relief-f* and the proposed *KRAV* methods weight the features with a more discriminative order. This is given by the fact that these methods incorporate the labels information to rank the original input features. Regarding this, *Relief-f* finds more important the features related to the Speed feature to improve the I.P. classification (see Fig. 4.2(b)). By the other hand, *KRAV* finds more relevant the features related to the Distance as seen in Fig. 4.2(c). The difference between the obtained $\boldsymbol{\varrho}$ for these methods can be grounded by the fact that *KRAV* considers the high class unbalance for this classification problem, while the *Relief-f* gives the same importance to all the samples regardless the class membership.

It can be seen from Fig. 4.3(e) that the proposed *KRAV* method obtains the highest classification performance of 0,6807 with the lowest number of employed features. When analyzing individually the class performance, it can be seen that given the high unbalance, the Disoriented class obtains the lowest classification performance, which only spikes when the selection of some particular features is obtained(see Fig. 4.3(a)). For the Distracted and Exploring classes the highest classification performance is obtained, about 0,77 and 0,83, respectively. Remarking that these values are obtained with the selection of a small set of features. Lastly, for the Interested class it can be seen that the *VRA* method has the worst performance, which can be explained by the low relevance given to the features related to the Distance feature.

As seen, the obtained $\boldsymbol{X}_S$ for I.P. does not have any bin related to gaze information. This is caused by some errors and noise associated with the automatic gaze estimation method,

which discretize the head orientation into 8 directions, and in our dataset we verified that such quantization is highly imbalance since the constant flow of pedestrians is restricted to a diagonal path where the 1, 2, 5, 6 walking directions are very frequent, and the remaining almost negligible. Another problem is that the gaze estimation technique is highly dependent of the head orientation, but that does not guarantee to be an accurate estimation of the pedestrian view frustum. Moreover, the histograms with the most bins are the ones of Trajectory, Distance and Speed that correspond to $R = 3$ in our multi-scale descriptor. In this sense, we select these three histograms to conform $\boldsymbol{X}_R$ for I.P.

Analyzing individually the methods, it is seen that *VRA* obtains the lowest performance for I.P. (0,5761), because it does not take into account supervised information to rank the relevance of the input features. The results for *Relief-f* improve in both classification (0,6288) and percentage of used features because of the inclusion of the labels information to find $\boldsymbol{\varrho}$. Now, $KRAV - S$ improves the I.P. results because it considers the class unbalance, obtaining 0,6807. Given the non-linear embedding, $KRAV - E$ obtains the best results for both I.P.,0,7481. Lastly, the high $KRAV - R$ results demonstrate that selection of the subsets exposed in Fig. **4-4** are enough for a good classification, remarking that the percentage of used features is reduced from 21 % to 12 %.

5. Kernel-based analysis for tunning C parameter in SVM classifier

Kernel-based estimation techniques, such as Support Vector Machines (SVMs) has shown to be powerful nonlinear classification. The method builds a linear model in the so-called feature space where the inputs have been transformed using a (possibly infinite dimensional) nonlinear mapping $\boldsymbol{\varphi}$. This is converted to the dual space using the Mercer's theorem and the use of a positive definite kernel, without computing the mapping explicitly $\boldsymbol{\varphi}$. Though the standard SVM owns better generalization performance compared with many other machine learning methods, the training stage of SVM involves solving a quadratic programming problem (QP) and is thus time-consuming. Its computational complexity is $O(m^3)$, where m is the total size of training points. This drawback restricts its application to some real problems [11, 16, 50]. Specifically, the video data expose high dimensional input spaces. Accordingly, tuning the SVMs parameters in a conventional way, by searching the parameter values over a grid to find the ideal ones. This increases the computational cost and makes the training stage counterproductive. We propose a Kernel-based analysis for estimating the box constraint parameter C in an SVM with RBF kernel. The proposal was taking advantage of the nonlinear relationships in the data and ideal relationships to determine an optimal value of C for each sample in the training stages. The methodology is tested on many datasets with different complexities.

5.1. Tuning C in SVM by Kernel-based analysis (K-SVM)

SMVs start with the goal of separating the data with a hyperplane and extend this to nonlinear decision boundaries using a kernel representations. The standard framework for SVMs estimation is based on a primal-dual formulation [16]. Given the dataset $\{\boldsymbol{x}_i, y_i\}_{i=1}^{M}$ with M samples, the goal is to estimate a model of the form

$$y_i(\boldsymbol{w}^\top \boldsymbol{\varphi}(\boldsymbol{x}_i) + b) \geq 1 - \xi_i \qquad \xi_i \geq 0 \tag{5-1}$$

where $\boldsymbol{x}_i \in \mathbb{R}^n$, $y_i \in \{-1, 1\}$, $\boldsymbol{w} \in \mathbb{R}^n$ and $\boldsymbol{\varphi}(\cdot) : \mathbb{R}^n \to \mathbb{R}^{n_h}$ is the mapping to a dimensional (and possibly infinite dimensional) feature space. The following optimization problem is formulated

$$\min \quad \frac{1}{2}\boldsymbol{w}^\top \boldsymbol{w} + C\sum_{i=1}^{M}\xi_i \tag{5-2}$$

$$s.t. \quad y_i(\boldsymbol{w}^\top \boldsymbol{\varphi}(\boldsymbol{x}_i) + b) \geq 1 - \xi_i \qquad \xi_i \geq 0 \tag{5-3}$$

The constrained permits a functional margin that is less than 1, and contains a penalty of cost $C\xi_i$, for any data point that falls within the margin on the correct side of the separating hyperplane (i.e., when $\xi_i > 1$). We thus state a preference for margins that classify the training data correctly but soften the constraints to allow for non-separable data with a penalty proportional to the amount by which the example is misclassified. Reformulating as a Lagrangian, which as before we need to minimize with respect to $\boldsymbol{w}$, b and ξ_i and maximize with respect to $\boldsymbol{\alpha}$ (where $\alpha_i \geq 0$, $\mu_i \geq 0 \ \ \forall_i$):

$$L_P \equiv \frac{1}{2}\|\boldsymbol{w}\|^2 + C\sum_{i=1}^{M}\xi_i - \sum_{i=1}^{M}\alpha_i[\ y_i(\boldsymbol{w}^\top \boldsymbol{\varphi}(\boldsymbol{x}_i) + b) - 1 + \xi_i] - \sum_{i=1}^{M}\mu_i\xi_i \tag{5-4}$$

differentiating with respect to $\boldsymbol{w}$, b and ξ_i and setting the derivatives to zero:

$$\frac{\partial L_P}{\partial \boldsymbol{w}} = 0 \Rightarrow \boldsymbol{w} = \sum_{i=1}^{M}\alpha_i y_i \boldsymbol{\varphi}(\boldsymbol{x}_i) \tag{5-5}$$

$$\frac{\partial L_P}{\partial b} = 0 \Rightarrow \sum_{i=1}^{M}\alpha_i y_i = 0 \tag{5-6}$$

$$\frac{\partial L_P}{\partial \xi_i} = 0 \Rightarrow C = \alpha_i + \mu_i \tag{5-7}$$

Substituting Eq.5-5 and Eq.5-6 into Eq.5-4 gives a new formulation which, begin dependent on $\boldsymbol{\alpha}$. However Eq.5-7 together with $\mu_i \geq 0 \ \ \forall_i$, implies that $\alpha \leq C$.

$$L_D \equiv \sum_{i=1}^{M}\alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i\alpha_j y_i y_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^\top \boldsymbol{\varphi}(\boldsymbol{x}_j) \quad s.t. \quad 0 \leq \alpha_i \leq C \ \ \forall_i \quad \text{and} \quad \sum_{i=1}^{M}\alpha_i y_i = 0 \tag{5-8}$$

$$\equiv \sum_{i=1}^{M}\alpha_i - \frac{1}{2}\sum_{i,j}\alpha_i H_{ij}\alpha_j \quad \text{where} \quad H_{ij} \equiv y_i y_j \boldsymbol{\varphi}(\boldsymbol{x}_i)^\top \boldsymbol{\varphi}(\boldsymbol{x}_j) \tag{5-9}$$

$$\equiv \sum_{i=1}^{M}\alpha_i - \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{H}\boldsymbol{\alpha} \quad s.t. \quad 0 \leq \alpha_i \leq C \ \ \forall_i \quad \text{and} \quad \sum_{i=1}^{M}\alpha_i y_i = 0 \tag{5-10}$$

we need to maximize

$$\max_{\alpha} \left[\sum_{i=1}^{M}\alpha_i - \frac{1}{2}\boldsymbol{\alpha}^\top \boldsymbol{H}\boldsymbol{\alpha}\right] \quad s.t. \ \ 0 \leq \alpha_i \leq C \ \ \forall_i \quad \text{and} \quad \sum_{i=1}^{M}\alpha_i y_i = 0 \tag{5-11}$$

with the application of the Mercer's theorem on the kernel matrix $\boldsymbol{K} \in \mathbb{R}^{M\times M}$ as $\boldsymbol{K}_{ij} = \boldsymbol{\kappa}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \boldsymbol{\varphi}(\boldsymbol{x}_i)^\top \boldsymbol{\varphi}(\boldsymbol{x}_j), i, j = 1, \cdots, M$. It is not required to compute explicitly the non-linear mapping $\boldsymbol{\varphi}(\cdot)$ as this done implicitly through the use of a positive definite kernel functions $\boldsymbol{K}$. We choose a Gaussian kernel due to the Gaussian function has the advantages of finding Hilbert spaces with universal approximating capability and its mathematical tractability [32], $\boldsymbol{K}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \exp(-\|\boldsymbol{x}_i - \boldsymbol{x}_j\|_2^2/2\sigma^2)$, where $\sigma \in \mathbb{R}^+$ is the kernel bandwidth. The parameter C in Eq.5-11 controls the relative weighting between the goal of making the margin small and ensuring that most examples have functional margins that are at least 1. We propose tuning the C parameter based on a distance between a gaussian kernel and a target kernel. The gaussian kernel keeps the nonlinear relationships among the training samples while a target kernel is an ideal kernel $\boldsymbol{K}_y$ that holds the pairwise similarity between labels y_i, $y_{i'}$. The distance d is calculated for each sample as

$$d_i = \frac{1}{M}\sum_{j=1}^{M} \left(\boldsymbol{K}(\boldsymbol{x}_i, \boldsymbol{x}_j; \sigma) - \boldsymbol{K}_y(y_i, y_j)\right)^2, \quad i = 1, \cdots, M \tag{5-12}$$

where $\boldsymbol{K}_y(y_i, y_j) = \delta_{y_i y_j}$ and delta function is $\delta_{y_i y_j} = 1$, if the samples have the same label $y_i = y_j$, otherwise $\delta_{y_i y_j} = 0$. This measure of distance d gives information about the similarity of each sample concerning the others. A value of $d_i = 0$ means a maximum similarity between the sample i and the others, while a value of $d_i = 1$ means that do not exist any similarity between the sample i regarding others. On this basis, we propose a function capable of penalizing in a strong manner the samples that show the lowest similarity regarding the samples of its same class as:

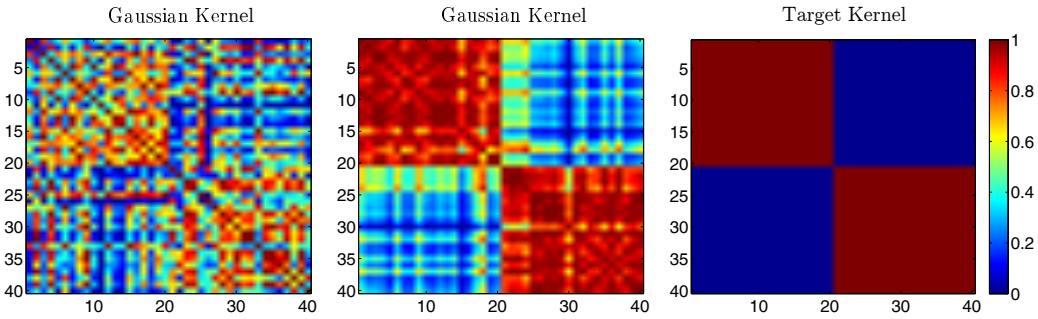
$$f(\boldsymbol{x}_i) = \exp\left(-\frac{d_i}{\sigma}\right) \tag{5-13}$$

The kernel bandwidth σ controls the observation window within the similarity distance is assessed. Finally, we set a different value of C parameter for each α_i value. A large value of C_i gives more weight to the sample i, this means that the optimization becomes more strict for that sample. Equivalently, a small value of C_i makes misclassification of that sample less important. Then, C is a vector of upper bounds in the optimization problem in Eq.5-11. Each value for C_i is computed as:

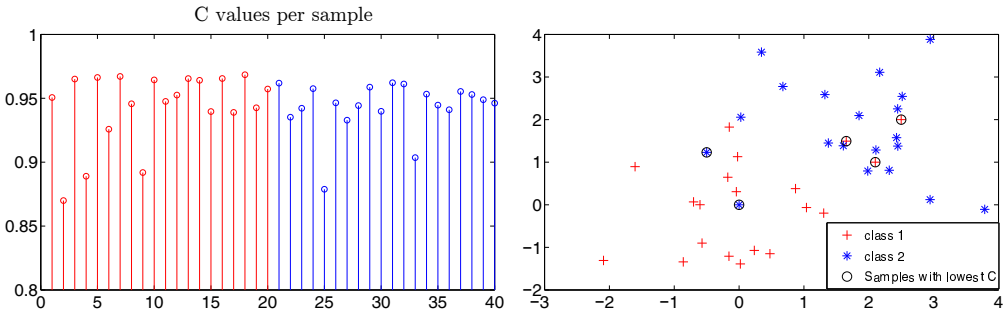
$$C_i = k_c f(\boldsymbol{x}_i) \tag{5-14}$$

The value of k_c is a scalar, it is computed as $m/(2m_1)$ for the data points of the group one and $m/(2m_2)$ for the data points of group two, where m_1 is the number of samples in group one, m_2 is the number of elements in group two, and $m = m_1 + m_2$. This rescaling is done to take into account unbalanced groups, that is cases where m_1 and m_2 have very different values.

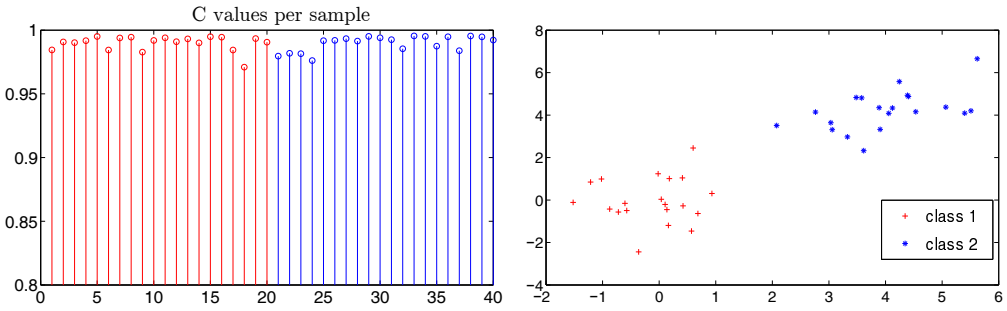
5.2. K-SVM experiments



(a) Kernel for test 1($\boldsymbol{K}_1$) (b) Kernel for test 2($\boldsymbol{K}_2$) (c) Target kernel($\boldsymbol{K}_L$)



(d) Samples for test 1



(e) Samples for test 2

Figure 5-1.: K-SVM: Estimating values of C from distance between $\boldsymbol{K}$ and $\boldsymbol{K}_L$

The methodology for tuning C parameter is summarized in Figure **5-1**. In this part, two random toy datasets are generated as illustrated in Figures 5.1(d) and 5.1(e). The first dataset exhibits some samples of class 1 overlapped with samples of class 2 while the second there is not overlapped samples. The similarity between both kernels $\boldsymbol{K}_1$ and $\boldsymbol{K}_L$ is lower than $\boldsymbol{K}_2$ and $\boldsymbol{K}_L$. Thus, the overlapped samples are penalized with a low C value as illustrated in Figure 5.1(d) in comparison with the samples of test 2 where all have a similar value of C as shown in Figure 5.1(e).

To test the K-SVM performance, we used three different ways to compute C. The first method uses a constant value C for all samples, and we set $C = 1$. The second uses different

values of C per class as does the constant k_c in Equation 5-14. The last one, it calculates the values of C with our proposal. Several datasets with different complexity to classification task are used. The complexity measure is based on the number of unbalanced classes in the dataset and the overlap between them. The Table **5-1** shown a summary of databases.

Dataset	# of samples	#classes	#attributes	complexity
dna	2000	3	180	low
glass	214	6	13	low
segment	2310	7	19	low
vowel	528	11	10	medium
connect 4	2869	3	125	high
shuttle	43500	7	9	high
Caviar	58	3	112	high
IIT-Shopping	1830	4	112	high

Table 5-1.: Datasets for testing K-SVM

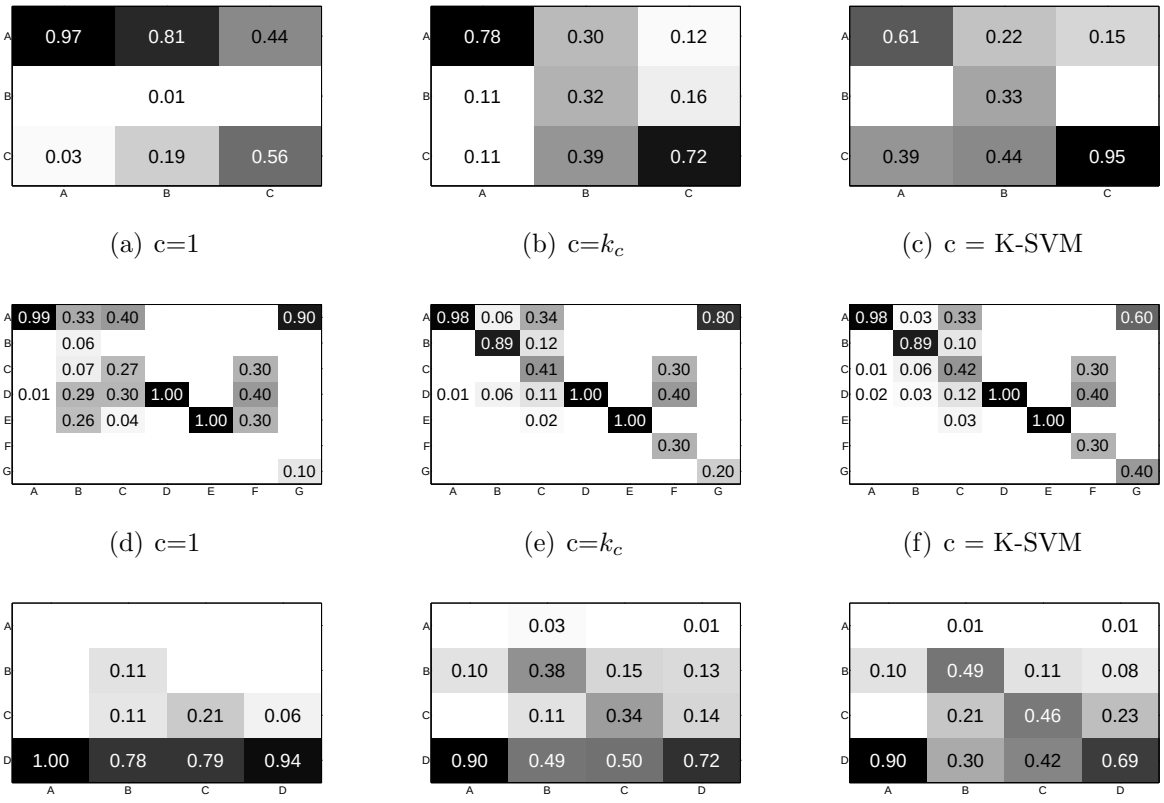
The criterions for evaluating the performance of the three methods are the supervised measures exposed in Section 3.2, precision (p), recall (r), $F1$ and accuracy (Acc). In some cases, the confusion matrix of classification is shown for presenting advantages of our method which are not exposed by the above measures. For training the SVMs models, we used a hold-out cross-validation framework with 70 % for training and 30 % for testing except ITT dataset because the class disoriented only has four samples. In this case, 75 % and 25 % are set for training and testing respectively. The experiments were run for 30 times to estimate the stability of the methods.

Dataset	$C=1$				$C=k_c$				C ours			
	r	p	F1	Acc	r	p	F1	Acc	r	p	F1	Acc
dna	95.06	95.07	95.06	96.71±1.23	95.18	95.21	95.18	96.78±1.31	**95.23**	**95.25**	**95.23**	**96.82±1.12**
glass	73.85	74.27	73.90	82.57±2.34	76.52	76.72	76.57	84.34±2.41	**78.60**	**87.81**	**78.66**	**85.73±2.17**
segment	88.46	89.25	88.27	96.70±1.59	88.43	89.64	88.69	96.69±1.62	**88.87**	**90.92**	**89.47**	**96.82±1.22**
vowel	77.54	78.71	76.19	95.91±2.87	83.87	**85.01**	83.95	97.06±3.01	**84.25**	84.92	84.21	**97.13±2.67**
connect 4	51.09	61.74	41.38	67.39±3.31	**60.26**	59.16	58.35	73.50±3.43	60.21	**71.63**	**59.09**	**74.03±3.13**
shuttle	48.79	56.07	37.18	85.37±4.23	68.23	78.79	64.81	90.92±3.21	**71.16**	**79.28**	**68.94**	**91.76±2.87**
Caviar	74.08	74.44	73.65	82.72±2.56	75.65	75.47	75.43	83.77±2.64	**77.19**	**77.48**	**77.12**	**84.80±2.38**
IIT-Shopping	31.53	**44.46**	23.04	65.76±1.67	36.08	33.80	31.51	68.04±1.26	**41.00**	36.11	**36.37**	**70.50±1.31**

Table 5-2.: Results of classification for the three SVM models

5.3. Discussion

Table **5-2** shows the clasification results on the dataset used for testing the K-SVM methodology. For the first four datasets (dna, glass, segment and vowel), the results exhibit similar



(a) c=1 (b) c=k_c (c) c = K-SVM

(d) c=1 (e) c=k_c (f) c = K-SVM

(g) c=1 (h) c=k_c (i) c = K-SVM

Figure 5-2.: Confusion matrices: the 1^{st} row are the results of Connect4 dataset, 2^{nd} row are the classification results of shuttle dataset and last row are the results for IIT dataset

performance since these one do not show marked imbalances in their classes, further, the overlap between the classes are small. These issues make the constant k_c (Eq. 5-14) take values close to 1 and therefore the values of C are also close to 1 for the three methodologies. Our method gets slight improvements as a result of it misclassifies the samples that expose a low similarity on the current class giving to SVM model more generality of classification. When our methodology is tested in connect4 dataset, it shows clearly the influence of tuning the C parameter properly. When this parameter is set equal for all classes, the class two is absorbed by the others. It may be since this class has fewer samples and skews the SVM model to the class with major number of samples (class one) as illustrated in Figure 5.2(a). When the C parameter is set taking into account the imbalance exposes by the classes, the class two improves its accuracy rate (32 %), due to, the C parameter takes the highest value for class two and the lowest value for the class one as illustrated in Figure 5.2(b). When the C parameter is tuned using K-SVM, this only does not take in considering the imbalanced classes, but also the separability between the classes is maximized as is shown in Figure 5.2(c) and the results in Table **5-2**. K-SVM obtains good performace when tested on shuttle dataset. This dataset has a large of samples as shown in Table 1. Classes 2, 3, 5 and 6 are classes with fewer samples and are highly correlated as illustred in Figure 5.2(d). K-SVM minimizes these correlations to maximize the class separability as shown in Figure 5.2(f). Class 6 went from 90 % FN rate with respect to class 1 to 60 %, achieving a high accuracy rate as shown in Table **5-2**. For IIT dataset, K-SVM improves the class separability between the class 3 and 4 (See Figure 5.2(i)). The results for class 1 are too low due to this one has four samples and not enough to establish a Kernel of similarities suitable. K-SVM obtains better classification rates than the others methods as seen in Table **5-2**.

Part IV.

Final Remarks

6. Conclusions and future work

6.1. Conclusions

In the present work, we have studied the use of kernel-based representations to support the analysis of video data by disclosing the intrinsic nonlinear relationships in the data. In this sense, three kernel strategies were proposed to reveal the salient features and relevant data for improving spatiotemporal descriptor representation, for supporting feature relevance analysis stages and support the training stages. The main concluding remarks are described below:

- We proposed a supervised kernel-based representation approach that allows improving the BoF representations for human activity recognition. Indeed, the introduced methodology, named CKA-BFR, includes prior knowledge to reveal the salient features to represent in a better way a pre-built codebook. So, CKA-BFR finds the intraclass relations that maximize the class separability for enhancing the classification performance. For such a purpose, CKA-BFR computes a mapping matrix based on a CKA-based function to force that initial relationships in the data keep an ideal behavior to a kernel target based on a vector of true labels. For the sake of comparison, CKA-BFR is tested as kernel data representation to improve BoF representations based on spatiotemporal features and BoFs based on trajectories of features. Attained results showed that CKA-BFR can find the best relations in a codebook to describe the periodic and complex human activities achieved an excellent performance regarding the classification metrics in comparison to the baseline methodologies to improve the BoF representations.

- We proposed a kernel-based relevance analysis methodology, named KRAV, as a tool for feature selection and extraction. KRAV learns a kernel matrix based on CKA approach taking advantage of the available joint information associating the employed trajectories of features with the corresponding social behavior labels. Afterwards, KRAV learns all relevant patterns from the input features, which are further used to compute new spaces where the class inter-separability is improved. Obtained results show that the selected relevant features and the nonlinear mapping improve the classification stage, and unlike commonly used relevance methods, KRAV handles in a better way class unbalance problems.

- We introduced a kernel-based analysis methodology for tuning the C parameter in an

SVM. In fact, the proposed strategy, named K-SVM, taking into account the similarity relationships in the data to penalize outliers samples regarding a determinate class. For this end, K-SVM computes a matrix distance between a kernel of initial similarities and an ideal kernel. Besides, K-SVM penalizes in an aggressive way to the samples with lowed similarity, and they are ignored in an optimization problem. Regarding K-SVM results, tuning the C parameter based on relation intra and inter-classes improving the SVM model and achieve high classification rates. Further, K-SVM is able to handle with the imbalanced class problem and able to generate a more generally model to handle with high correlations between the classes.

6.2. Future work

From the attained results and the drawbacks found along the process, the following theoretical and experimental topics could be explored:

- CKA-BFR was proposed to improve BoF representation generated by k-means. As future work, we can measure the impact of our method when the BoF representation is generated by more elaborate methods such as models based on GMM, Fisher vector encoding among other presents in the state-of-the-art.

- KRAV only was tested to determinate the relevance of social behavior descriptors on IIT-shopping dataset. As future work, the KRAV analysis can be tested on other real dataset using the same set of social behavior descriptors for comparing the results obtained in the first datasets and measure the real impact of each descriptor.

- K-SVM was presented for tuning the C parameter in SVM. Specifically, the methodology was tested in RBF-SVM. As future work, K-SVM can be extend to SVM models with different kernel such as polynomial kernels, sigmoid kernels among others.

- The proposed Kernel-based framework finds similarity relations in the data, using Gaussian kernel. As future work, the framework can be extend to compute similarities with other families of kernel regarding the specific task.

Bibliography

[1] Adam, Amit ; Rivlin, Ehud ; Shimshoni, Ilan ; Reinitz, David: Robust real-time unusual event detection using multiple fixed-location monitors. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 30 (2008), Nr. 3, S. 555–560 21

[2] Aggarwal, Jake K. ; Ryoo, Michael S.: Human activity analysis: A review. In: *ACM Computing Surveys (CSUR)* 43 (2011), Nr. 3, S. 16 5, 6

[3] Álvarez-Meza, AM ; Cárdenas-Peña, D ; Castellanos-Dominguez, Germán: Unsupervised kernel function building using maximization of information potential variability. In: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer, 2014, S. 335–342 22, 30

[4] Alvarez-Meza, Andres M. ; Daza-Santacoloma, Genaro ; Castellanos-Dominguez, German: Biomedical data analysis by supervised manifold learning. In: *Engineering in Medicine and Biology Society (EMBC), 2012 Annual International Conference of the IEEE* IEEE, 2012, S. 41–44 3

[5] Alvarez-Meza, Andres M. ; Velasquez-Martinez, LF ; Castellanos-Dominguez, German: Feature relevance analysis supporting automatic motor imagery discrimination in EEG based BCI systems. In: *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE* IEEE, 2013, S. 7068–7071 3

[6] Bay, Herbert ; Ess, Andreas ; Tuytelaars, Tinne ; Van Gool, Luc: Speeded-up robust features (SURF). In: *Computer vision and image understanding* 110 (2008), Nr. 3, S. 346–359 6

[7] Blank, Moshe ; Gorelick, Lena ; Shechtman, Eli ; Irani, Michal ; Basri, Ronen: Actions as space-time shapes. In: *Computer Vision, 2005. ICCV 2005. Tenth IEEE International Conference on* Bd. 2 IEEE, 2005, S. 1395–1402 20

[8] Brockmeier, Austin J. ; Choi, John S. ; Kriminger, Evan G. ; Francis, Joseph T. ; Principe, Jose C.: Neural decoding with kernel-based metric learning. In: *Neural computation* 26 (2014), Nr. 6, S. 1080–1107 19, 28

[9] Camps-Valls, Gustavo ; Bruzzone, Lorenzo u. a.: *Kernel methods for remote sensing data analysis*. Bd. 2. Wiley Online Library, 2009 3

[10] Cortes, Corinna ; Mohri, Mehryar ; Rostamizadeh, Afshin: Algorithms for learning kernels based on centered alignment. In: *The Journal of Machine Learning Research* 13 (2012), Nr. 1, S. 795–828 19

[11] Cristianini, Nello ; Shawe-Taylor, John: *An introduction to support vector machines*. 2000 36

[12] Dalal, Navneet: *Finding people in images and videos*, Institut National Polytechnique de Grenoble-INPG, Diss., 2006 5

[13] Dalal, Navneet ; Triggs, Bill: Histograms of oriented gradients for human detection. In: *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on* Bd. 1 IEEE, 2005, S. 886–893 5

[14] Daza-Santacoloma, Genaro ; Arias-Londoño, Julián D ; Godino-Llorente, Juan I. ; Sáenz-Lechón, Nicolás ; Osma-Ruíz, Víctor ; Castellanos-Domínguez, Germán: Dynamic feature extraction: an application to voice pathology detection. In: *Intelligent Automation & Soft Computing* 15 (2009), Nr. 4, S. 667–682 30

[15] Dollár, Piotr ; Rabaud, Vincent ; Cottrell, Garrison ; Belongie, Serge: Behavior recognition via sparse spatio-temporal features. In: *Visual Surveillance and Performance Evaluation of Tracking and Surveillance, 2005. 2nd Joint IEEE International Workshop on* IEEE, 2005, S. 65–72 4, 6

[16] Espinoza, Marcelo ; Suykens, Johan A. ; De Moor, Bart: Fixed-size least squares support vector machines: A large scale application in electrical load forecasting. In: *Computational Management Science* 3 (2006), Nr. 2, S. 113–129 36

[17] García-Vega, Sergio ; Álvarez-Meza, Andrés M. ; Castellanos-Domínguez, César G.: MoCap Data Segmentation and Classification Using Kernel Based Multichannel Analysis. In: *Progress in Pattern Recognition, Image Analysis, Computer Vision, and Applications*. Springer, 2013, S. 495–502 3

[18] Gretton, Arthur ; Bousquet, Olivier ; Smola, Alex ; Schölkopf, Bernhard: Measuring statistical dependence with Hilbert-Schmidt norms. In: *Algorithmic learning theory* Springer, 2005, S. 63–77 28

[19] Gurwicz, Yaniv ; Yehezkel, Raanan ; Lachover, Boaz: Multiclass object classification for real-time video surveillance systems. In: *Pattern Recognition Letters* 32 (2011), Nr. 6, S. 805–815 4

[20] Hall, Mark A.: *Correlation-based feature selection for machine learning*, The University of Waikato, Diss., 1999 7

[21] Honeine, Paul: Online kernel principal component analysis: A reduced-order model. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34 (2012), Nr. 9, S. 1814–1826 7

[22] John, George H. ; Kohavi, Ron ; Pfleger, Karl u. a.: Irrelevant features and the subset selection problem. In: *Machine Learning: Proceedings of the Eleventh International Conference*, 1994, S. 121–129 7

[23] Jones, Simon ; Shao, Ling ; Zhang, Jianguo ; Liu, Yan: Relevance feedback for real-world human action retrieval. In: *Pattern Recognition Letters* 33 (2012), Nr. 4, S. 446–452 3

[24] Kachouane, M ; Sahki, S ; Lakrouf, M ; Ouadah, N: HOG based fast human detection. In: *Microelectronics (ICM), 2012 24th International Conference on* IEEE, 2012, S. 1–4 5

[25] Kecman, Vojislav: *Learning and soft computing: support vector machines, neural networks, and fuzzy logic models*. MIT press, 2001 7

[26] Klaser, Alexander ; Marszałek, Marcin ; Schmid, Cordelia: A spatio-temporal descriptor based on 3d-gradients. In: *BMVC 2008-19th British Machine Vision Conference* British Machine Vision Association, 2008, S. 275–1 6

[27] Kreyszig, Erwin: *Introductory functional analysis with applications*. Bd. 81. wiley New York, 1989 11

[28] Laptev, Ivan: On space-time interest points. In: *International Journal of Computer Vision* 64 (2005), Nr. 2-3, S. 107–123 6

[29] Laptev, Ivan ; Lindeberg, Tony: Local descriptors for spatio-temporal recognition. In: *Spatial Coherence for Visual Motion Analysis*. Springer, 2006, S. 91–103 6

[30] Laptev, Ivan ; Marszałek, Marcin ; Schmid, Cordelia ; Rozenfeld, Benjamin: Learning realistic human actions from movies. In: *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on* IEEE, 2008, S. 1–8 6

[31] Liu, Jia ; Wang, Xiaonian ; Li, Tianyu ; Yang, Jie: Spatio-temporal Semantic Features for Human Action Recognition. In: *KSII Transactions on Internet and Information Systems (TIIS)* 6 (2012), Nr. 10, S. 2632–2649 7

[32] Liu, Weifeng ; Principe, Jose C. ; Haykin, Simon: *Kernel adaptive filtering: a comprehensive introduction*. Bd. 57. John Wiley & Sons, 2011 38

[33] Lopez-Villa, JS ; Insuasti-Ceballos, HD ; Molina-Giraldo, S ; Alvarez-Meza, A ; Castellanos-Dominguez, G: A novel tool for ground truth data generation for video-based object classification. (2015) 3

[34] Lowe, David G.: Distinctive image features from scale-invariant keypoints. In: *International journal of computer vision* 60 (2004), Nr. 2, S. 91–110 6

[35] Lu, Wei-Lwun ; Little, James J.: Simultaneous tracking and action recognition using the pca-hog descriptor. In: *Computer and Robot Vision, 2006. The 3rd Canadian Conference on* IEEE, 2006, S. 6–6 7

[36] Lu, Yanyun ; Boukharouba, Khaled ; Boonært, Jacques ; Fleury, Anthony ; Lecoeuche, Stéphane: Application of an incremental SVM algorithm for on-line human recognition from video surveillance using texture and color features. In: *Neurocomputing* 126 (2014), S. 132–140 4, 7

[37] Molina-Giraldo, S ; Insuasti-Ceballos, HD ; Arroyave, CE ; Montoya, JF ; Lopez-Villa, JS ; Alvarez-Meza, A ; Castellanos-Dominguez, G: People detection in video streams using background subtraction and spatial-based scene modeling. (2015) 3

[38] Noguchi, Akitsugu ; Yanai, Keiji: A surf-based spatio-temporal feature for feature-fusion-based action recognition. In: *Trends and Topics in Computer Vision*. Springer, 2012, S. 153–167 6, 7

[39] Oussalah, Mourad ; Professor Ali Hessami, Dr ; M'hamed Abidine, Bilal ; Fergani, Belkacem ; Oussalah, Mourad ; Fergani, Lamya: A new classification strategy for human activity recognition using cost sensitive support vector machines for imbalanced data. In: *Kybernetes* 43 (2014), Nr. 8, S. 1150–1164 8

[40] Parzen, Emanuel: *Statistical inference on time series by Hilbert space methods*. Stanford Univ., 1959 13

[41] Peng, Xiaojiang ; Wang, Limin ; Wang, Xingxing ; Qiao, Yu: Bag of visual words and fusion methods for action recognition: Comprehensive study and good practice. In: *arXiv preprint arXiv:1405.4506* (2014) 4, 5, 17

[42] Pereira, Eduardo M. ; Ciobanu, Lucian ; Cardoso, Jaime S.: Context-based trajectory descriptor for human activity profiling. In: *Systems, Man and Cybernetics (SMC), 2014 IEEE International Conference on* IEEE, 2014, S. 2385–2390 7, 20, 21, 24, 29

[43] Popoola, Oluwatoyin P. ; Wang, Kejun: Video-based abnormal human behavior recognition—a review. In: *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 42 (2012), Nr. 6, S. 865–878 3

[44] Poppe, Ronald: A survey on vision-based human action recognition. In: *Image and vision computing* 28 (2010), Nr. 6, S. 976–990 5, 17

[45] Ramírez-Giraldo, Daniela ; Molina-Giraldo, Santiago ; Álvarez-Meza, Andrés M ; Daza-Santacoloma, Genaro ; Castellanos-Domínguez, Germán: Kernel based hand gesture recognition using kinect sensor. In: *Image, Signal Processing, and Artificial Vision (STSIVA), 2012 XVII Symposium of* IEEE, 2012, S. 158–161 3

[46] Ribeiro, Pedro C. ; Santos-Victor, José: Human activity recognition from video: modeling, feature selection and classification architecture. In: *Proceedings of International Workshop on Human Activity Recognition and Modelling* Citeseer, 2005, S. 61–78 4, 7

[47] Robnik-Šikonja, Marko ; Kononenko, Igor: Theoretical and empirical analysis of ReliefF and RReliefF. In: *Machine learning* 53 (2003), Nr. 1-2, S. 23–69 30

[48] Sadanand, Sreemananth ; Corso, Jason J.: Action bank: A high-level representation of activity in video. In: *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on* IEEE, 2012, S. 1234–1241 5, 6, 7, 17

[49] Schölkopf, Bernhard ; Smola, Alexander ; Müller, Klaus-Robert: Nonlinear component analysis as a kernel eigenvalue problem. In: *Neural computation* 10 (1998), Nr. 5, S. 1299–1319 7

[50] Scholkopf, Bernhard ; Smola, Alexander J.: *Learning with kernels: support vector machines, regularization, optimization, and beyond*. MIT press, 2001 14, 36

[51] Schüldt, Christian ; Laptev, Ivan ; Caputo, Barbara: Recognizing human actions: a local SVM approach. In: *Pattern Recognition, 2004. ICPR 2004. Proceedings of the 17th International Conference on* Bd. 3 IEEE, 2004, S. 32–36 7, 20

[52] Shao, Yuan-Hai ; Chen, Wei-Jie ; Zhang, Jing-Jing ; Wang, Zhen ; Deng, Nai-Yang: An efficient weighted Lagrangian twin support vector machine for imbalanced data classification. In: *Pattern Recognition* 47 (2014), Nr. 9, S. 3158–3167 8

[53] Shawe-Taylor, John ; Cristianini, Nello: *Kernel methods for pattern analysis*. Cambridge university press, 2004 3

[54] Suykens, Johan A. ; De Brabanter, Jos ; Lukas, Lukas ; Vandewalle, Joos: Weighted least squares support vector machines: robustness and sparse approximation. In: *Neurocomputing* 48 (2002), Nr. 1, S. 85–105 8

[55] Suykens, Johan A. ; Vandewalle, Joos: Least squares support vector machine classifiers. In: *Neural processing letters* 9 (1999), Nr. 3, S. 293–300 7

[56] Takahashi, Masaki ; Naemura, Masahide ; Fujii, Mahito ; Satoh, Shin'ichi: Human action recognition in crowded surveillance video sequences by using features taken from key-point trajectories. In: *Computer Vision and Pattern Recognition Workshops (CVPRW), 2011 IEEE Computer Society Conference on* IEEE, 2011, S. 9–16 6, 7, 23

[57] Tian, YingLi ; Cao, Liangliang ; Liu, Zicheng ; Zhang, Zhengyou: Hierarchical filtered motion for action recognition in crowded videos. In: *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on* 42 (2012), Nr. 3, S. 313–323 17

[58] Tseng, Chien-Chung ; Chen, Ju-Chin ; Fang, Ching-Hsien ; Lien, Jenn-Jier J.: Human action recognition based on graph-embedded spatio-temporal subspace. In: *Pattern Recognition* 45 (2012), Nr. 10, S. 3611–3624 7

[59] Twining, Carole J. ; Taylor, Christopher J.: The use of kernel principal component analysis to model data distributions. In: *Pattern Recognition* 36 (2003), Nr. 1, S. 217–227 7

[60] Velásquez-Martínez, Luisa F. ; Álvarez-Meza, Andrés M. ; Castellanos-Domínguez, César G.: Motor imagery classification for bci using common spatial patterns and feature relevance analysis. In: *Natural and Artificial Computation in Engineering and Medical Applications*. Springer, 2013, S. 365–374 3

[61] Vishwakarma, Sarvesh ; Agrawal, Anupam: A survey on activity recognition and behavior understanding in video surveillance. In: *The Visual Computer* 29 (2013), Nr. 10, S. 983–1009 6

[62] Wang, Heng ; Kläser, Alexander ; Schmid, Cordelia ; Liu, Cheng-Lin: Dense trajectories and motion boundary descriptors for action recognition. In: *International journal of computer vision* 103 (2013), Nr. 1, S. 60–79 6, 7, 17

[63] Wang, Heng ; Schmid, Cordelia: Action recognition with improved trajectories. In: *Computer Vision (ICCV), 2013 IEEE International Conference on* IEEE, 2013, S. 3551–3558 5, 6, 7

[64] Wang, Heng ; Ullah, Muhammad M. ; Klaser, Alexander ; Laptev, Ivan ; Schmid, Cordelia: Evaluation of local spatio-temporal features for action recognition. In: *BMVC 2009-British Machine Vision Conference* BMVA Press, 2009, S. 124–1 17

[65] Wang, Jinqiao ; Fu, Wei ; Lu, Hanqing ; Ma, Songde: Bilayer Sparse Topic Model for Scene Analysis in Imbalanced Surveillance Videos. In: *Image Processing, IEEE Transactions on* 23 (2014), Nr. 12, S. 5198–5208 5

[66] Wang, Liang ; Suter, David: Recognizing human activities from silhouettes: Motion subspace and factorial discriminative graphical model. In: *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on* IEEE, 2007, S. 1–8 7

[67] Wang, Wei ; Zhang, Peng ; Wang, Runsheng: Abnormal video sections detection based on inter-frames information. In: *Multimedia and Ubiquitous Engineering, 2009. MUE'09. Third International Conference on* IEEE, 2009, S. 529–533 6, 7

[68] Wang, Xiaogang: Intelligent multi-camera video surveillance: A review. In: *Pattern recognition letters* 34 (2013), Nr. 1, S. 3–19 3, 4

[69] Weinland, Daniel ; Ronfard, Remi ; Boyer, Edmond: A survey of vision-based methods for action representation, segmentation and recognition. In: *Computer Vision and Image Understanding* 115 (2011), Nr. 2, S. 224–241 5

[70] Willems, Geert ; Tuytelaars, Tinne ; Van Gool, Luc: An efficient dense and scale-invariant spatio-temporal interest point detector. In: *Computer Vision–ECCV 2008*. Springer, 2008, S. 650–663 6

[71] Xian, Yang ; Rong, Xuejian ; Yang, Xiaodong ; Tian, Yingli: CCNY at TRECVID 2014: Surveillance Event Detection. 4, 7

[72] Xu, Wanru ; Miao, Zhenjiang ; Zhang, Xiao-Ping: Structured feature-graph model for human activity recognition. In: *Image Processing (ICIP), 2015 IEEE International Conference on* IEEE, 2015, S. 1245–1249 7

[73] Yacoob, Yaser ; Black, Michael J.: Parameterized modeling and recognition of activities. In: *Computer Vision, 1998. Sixth International Conference on* IEEE, 1998, S. 120–127 6

[74] Yang, Xiaodong ; Yi, Chucai ; Cao, Liangliang ; Tian, Y: MediaCCNY at TRECVID 2012: Surveillance event detection. In: *NIST TRECVID Workshop*, 2012 4, 5, 7, 21

[75] Zhang, Chenyang ; Yang, Xiaodong ; Yi, Chucai ; Tian, Yingli ; Yu, Qian ; Tamrakar, Amir ; Divakaran, Ajay: CCNY-SRI@ TRECVid 2013 intED: a Human Interactive Event Detection System. 6

[76] Zhang, Zhang ; Tao, Dacheng: Slow feature analysis for human action recognition. In: *Pattern Analysis and Machine Intelligence, IEEE Transactions on* 34 (2012), Nr. 3, S. 436–450 7